International Journal of Engineering Research and Science & Technology

www.ijerst.org

ISSN : 2319-5991

Vol. 22 No. 2(1) (2026)



ijerst.editor@gmail.com

editor@ijerst.com

ISSN 2319-5991 www.ijerst.com
© 2026 IJERST. All Rights Reserved
Peer Reviewed, Referred & Indexed Journal

Research Paper

Recommendation Based AI Powered Medicine Expiry and Counterfeit Detection System

K. S. J. J. Manoj, N. Madhu Sai Latha, K. Jhansi, K. Sai Nagendra, Mr. M. V. Ramana Murthy

Department of Computer Science and Engineering (Data Science), Raghu Engineering College (Autonomous), Affiliated to JNTU Gurajada, Vizianagaram, Visakhapatnam, India

First Author: K. S. J. J Manoj | manojkandula2005@gmail.com

Corresponding Author: N. Madhu sai latha | madhunanepalli91@gmail.com

Guide: Mr. M. V. Ramana Murthy, Assistant Professor, Dept. of CSE (Data Science), Raghu Engineering College

Abstract

Counterfeit and expired medicines pose critical global health risks, with the World Health Organization estimating that 1 in 10 medical products in developing countries are substandard or falsified. Traditional manual inspection methods are time-consuming, error-prone, and fail to scale with pharmaceutical supply chain complexity. This paper presents **MediCheck**, an intelligent web-based pharmaceutical authentication system that integrates Optical Character Recognition (OCR), regex-based date extraction, and deep learning to automatically detect medicine expiry dates and identify counterfeit products in real time. The expiry detection module employs Tesseract OCR with dual page segmentation modes (PSM 3 and PSM 6), combined with a layered regex engine tailored to Indian pharmaceutical labelling formats as mandated by the Central Drugs Standard Control Organisation (CDSCO), including MM/YYYY, DD/MM/YYYY, and abbreviated month-year formats. The counterfeit detection module fine-tunes an EfficientNet-B0 convolutional neural network using a two-phase transfer learning strategy, achieving a validation accuracy of 99.78% on a curated dataset of real and fake medicine images. Both modules are integrated into a Flask-based responsive web application, enabling users to upload a medicine image and receive instant verification results. The proposed system demonstrates practical applicability for pharmacies, hospitals, distribution centres, and individual patients, offering a scalable solution for reducing medication errors and protecting public health.

Keywords: Optical Character Recognition, EfficientNet-B0, Counterfeit Medicine Detection, Expiry Date Extraction, Transfer Learning, Flask Web Application

1. Introduction

The authenticity and safety of pharmaceutical products are among the most critical concerns in global healthcare. Counterfeit medicines — defined as falsified products that deliberately misrepresent their identity, composition, or source

— and expired medicines that reach end consumers cause treatment failures, drug resistance, and in severe cases, fatalities. According to the World Health Organization (WHO), up to 10% of medicines circulating in low- and middle-income countries, including India, are substandard or falsified. In India specifically, the Central Drugs Standard Control Organization (CDSCO) identifies counterfeit and substandard drugs as a persistent challenge in the pharmaceutical market, with diverse packaging formats across manufacturers compounding the difficulty of verification.

Conventional approaches to medicine verification rely almost entirely on manual inspection. Pharmacists and patients must physically read printed labels to identify manufacturing and expiry dates, and assess packaging features such as colour consistency, logo placement, and seal integrity to detect counterfeits. These methods are not only labour-intensive but are also highly prone to human error, particularly when expiry dates are printed in small fonts, rendered unclear due to packaging damage, or presented in inconsistent regional formats. No existing widely deployed system combines expiry date detection and counterfeit identification in a single accessible platform for ordinary users.

This paper proposes MediCheck, a dual-module AI-powered system that addresses both challenges in a unified web application. The system employs Tesseract OCR with optimised preprocessing and regex pattern engineering for expiry date extraction and fine-tunes an EfficientNet-B0 deep learning model for fake-versus-real medicine classification. Built on a Flask backend with a responsive HTML/CSS/JavaScript interface, MediCheck is accessible to patients, pharmacists, healthcare workers, and regulatory personnel without requiring technical expertise. The primary objectives of this work are:

- To design and implement an automated expiry date detection system using Tesseract OCR and regex-based pattern recognition optimised for Indian pharmaceutical labelling standards.
- To design and implement a fake-versus-real medicine classification system using a deep learning model based on EfficientNet-B0, trained through transfer learning to distinguish genuine pharmaceutical products from counterfeits.
- To integrate both modules into a unified, user-friendly web application built with Flask, enabling real-time medicine verification through a clean and accessible interface.

The remainder of this paper is organised as follows. Section 2 reviews related literature. Section 3 presents the system design and architecture. Section 4 details the implementation methodology. Section 5 presents experimental results and discussion. Section 6 concludes the paper and outlines future work.

2. Literature Survey

2.1 OCR and Text Extraction from Pharmaceutical Packaging

Optical Character Recognition (OCR) has been widely studied for extracting text from real-world images, with pharmaceutical packaging representing one of its most safety-critical applications. Modern pharmaceutical labels present unique challenges including curved surfaces, embossed printing, low-contrast ink, and small font sizes. Researchers have demonstrated

that preprocessing techniques such as adaptive thresholding, Contrast Limited Adaptive Histogram Equalization (CLAHE), and bilateral filtering significantly improve OCR accuracy on degraded packaging images. Tesseract OCR, an open-source engine maintained by Google, has been widely adopted in pharmaceutical text extraction research due to its support for multiple Page Segmentation Modes (PSM) that accommodate different text layouts found on medicine labels [Smith, 2007].

2.2 Expiry Date Detection and Regex-Based Information Extraction

Regular expression (regex)-based approaches have been shown to be highly effective for extracting date entities from OCR output, particularly when the target formats follow domain-specific conventions. In the context of Indian pharmaceutical packaging, the CDSCO mandates specific labelling formats for manufacturing and expiry dates, including formats such as MM/YYYY, DD/MM/YYYY, and abbreviated month-year formats like JAN 2026. Studies on information extraction from medical packaging highlight the importance of designing regex patterns that account for OCR errors such as character substitutions and spacing inconsistencies — common artefacts of text extracted from printed labels [Chiticariu et al., 2013].

2.3 Counterfeit Medicine Detection Using Image Processing

The detection of counterfeit pharmaceutical products using image-based methods has received increasing attention from researchers and regulatory bodies. Early approaches relied on handcrafted image features such as colour histograms and texture descriptors, but these methods demonstrated limited generalisation across different medicine brands and packaging styles. The emergence of deep learning has opened new directions for counterfeit detection by enabling models to learn discriminative visual features automatically from labelled image datasets. The WHO estimates that up to 10% of medicines in low- and middle-income countries are substandard or falsified, making automated detection tools a public health priority [WHO, 2017].

2.4 Deep Learning and CNNs for Image Classification

Convolutional Neural Networks (CNNs) have established themselves as the dominant architecture for image classification since the landmark success of AlexNet in the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Subsequent architectures including VGGNet, ResNet, InceptionNet, and EfficientNet have progressively improved classification accuracy while reducing computational cost. EfficientNet, introduced by Tan and Le (2019), employs a compound scaling method that uniformly scales network width, depth, and resolution, achieving state-of-the-art accuracy with significantly fewer parameters. EfficientNet-B0, the baseline model in the EfficientNet family, has demonstrated strong performance on binary classification tasks including medical image analysis, making it well-suited for the fake-versus-real medicine detection task in this project [Tan & Le, 2019].

2.5 Transfer Learning in Medical and Pharmaceutical Image Analysis

Transfer learning has emerged as a powerful strategy for image classification tasks in domains where large labelled datasets are difficult to obtain, such as medical and pharmaceutical imaging. By

leveraging weights pretrained on large general-purpose datasets such as ImageNet, models can be fine-tuned on smaller domain-specific datasets while retaining rich feature representations. Studies in medical image classification have consistently demonstrated that transfer learning from ImageNet pretrained CNNs outperforms training from scratch, even when the source and target domains differ significantly. The two-phase fine-tuning strategy — freezing the backbone during initial training then fine-tuning the full network at a reduced learning rate — prevents catastrophic forgetting while allowing the model to adapt to domain-specific visual patterns [Tajbakhsh et al., 2016].

2.6 Web-Based Deployment of Deep Learning Models

Flask, a lightweight Python web framework, has been widely adopted for deploying machine learning models due to its simplicity, flexibility, and seamless integration with Python-based AI libraries such as PyTorch and TensorFlow. Research on model deployment emphasises the importance of decoupling the inference pipeline from the web serving layer, loading models once at application startup rather than per request, and using asynchronous JavaScript requests (AJAX/Fetch API) to deliver results without full page reloads. Studies also highlight the need for robust input validation and error handling in production deployments to ensure reliability when processing real-world user-uploaded images [Ketkar & Santana, 2021].

2.7 Challenges in Real-World Pharmaceutical Image Analysis

Literature consistently highlights several challenges specific to pharmaceutical image analysis. Variable lighting conditions during image capture, curved or reflective packaging surfaces, damage or smudging of printed text, and the wide diversity of medicine packaging designs across manufacturers all contribute to the difficulty of building robust automated systems. Studies on OCR applied to product packaging note that consumer-grade smartphone cameras, which represent the most realistic image capture scenario for end users, introduce additional challenges such as motion blur, perspective distortion, and uneven illumination. Addressing these challenges requires a combination of robust image preprocessing, ensemble OCR strategies using multiple page segmentation modes, and data augmentation during model training [Karatzas et al., 2015].

3. System Design

3.1 System Architecture Overview

The proposed MediCheck system is built on a modular, two-track architecture integrated within a single Flask-based web platform. Users interact with the system through a responsive web interface where they upload a medicine image and select the desired verification type — expiry date detection or fake-versus-real classification. Figure 1 illustrates the high-level system architecture.

The system consists of the following principal components:

- **Web Interface Module** — A responsive frontend built with HTML, CSS, and JavaScript enabling image upload and result display via asynchronous REST API calls.
- **Expiry Date Detection Module** — An OCR and regex pipeline for extracting manufacturing and expiry dates from medicine packaging.

- **Fake vs Real Detection Module** — A deep learning classification module using a fine-tuned EfficientNet-B0 model.
- **Flask Backend** — The application server coordinating routing, image handling, inference, and result serving.
- **Model and OCR Storage** — Trained model file (medicine_model_info.pth), OCR script (text_scrapper.py), and temporary uploaded image storage.

3.2 Expiry Date Detection Module Design

The expiry date detection pipeline begins with image acquisition through the Flask web interface. The uploaded image is preprocessed using OpenCV with adaptive contrast enhancement and noise reduction to improve OCR reliability on degraded or low-contrast packaging. Tesseract OCR is then applied in a dual-pass configuration: PSM 3 (fully automatic page segmentation) and PSM 6 (assume a single uniform block of text), with results from both passes merged to maximise text coverage across different label layouts.

The extracted text is passed through a layered regex engine containing patterns specifically engineered for Indian pharmaceutical date formats as mandated under CDSCO labelling guidelines. These include:

Pattern Type	Example Format
Numeric month-year	MM/YYYY (e.g., 03/2026)
Full date	DD/MM/YYYY (e.g., 15/03/2026)
Abbreviated month-year	MON YYYY (e.g., MAR 2026)
Compact format	MMYYYY (e.g., 032026)
Hyphenated date	DD-MON-YYYY (e.g., 15-MAR-2026)

After date identification, the system calculates remaining days to expiry or days since manufacture, producing a clear status message (e.g., VALID — 180 days remaining, or EXPIRED — 45 days ago) for the user.

3.3 Fake vs Real Detection Module Design

The fake-versus-real detection module employs a transfer learning approach using EfficientNet-B0 pretrained on ImageNet. The classification head is replaced with a custom sequential layer consisting of a Dropout layer ($p = 0.4$), a fully connected layer (512 units) with ReLU activation, a second Dropout layer ($p = 0.3$), and a final output layer with 2 units (real / fake). Training proceeds in two phases: Phase 1 (15 epochs) freezes the backbone and trains only the classification head; Phase 2 (15 epochs) unfreezes the full network and fine-tunes

at a reduced learning rate using the ReduceLROnPlateau scheduler. Input images are resized to 224 × 224 pixels and normalised using ImageNet mean and standard deviation values.

3.4 UML Diagrams

The system interaction is modelled through the following UML artefacts: a Use Case Diagram capturing user–system interaction for both detection flows; a Class Diagram defining the MedicineImage, ExpiryDetector, FakeRealDetector, Model, and Result classes with their attributes and methods; a Sequence Diagram tracing the message flow from user image upload through OCR/deep learning processing to result display; an Activity Diagram illustrating the decision flow for detection type selection, preprocessing, and result generation; and a Deployment Diagram mapping client device, web server, application server, OCR engine, date extraction module, deep learning model, and database to their physical infrastructure.

4. Implementation

4.1 Dataset Collection and Preprocessing

A diverse dataset of high-resolution medicine package and label images was collected from publicly available datasets, pharmacy samples, and manually captured images under different environmental conditions. The dataset encompasses medicine strips, bottles, and boxes with varied fonts, label layouts, packaging styles, camera angles, and lighting conditions. Separate categories are maintained for real and fake medicines.

Preprocessing steps applied:

- Resizing all images to 224 × 224 pixels for model input consistency
- Noise reduction and contrast normalisation for OCR preprocessing
- Data augmentation for model training: random horizontal/vertical flipping, rotation (up to ±30°), zoom (up to 20%), and brightness and contrast jitter to improve generalisation

The dataset was split into training (70%), validation (15%), and test (15%) sets. Class balance was maintained across all splits to prevent model bias.

4.2 Expiry Date Detection Implementation

The OCR pipeline is implemented in text_scrapper.py. The key processing steps are:

1. Image loaded via Flask and passed to the preprocessing function
2. OpenCV applies adaptive histogram equalisation and Gaussian blur reduction
3. pytesseract.image_to_string() is called twice with --psm 3 and --psm 6
4. Combined OCR output is passed to the regex engine
5. Matched dates are parsed using Python's datetime module
6. Days to expiry calculated from today's date; result returned as JSON to the frontend

4.3 Deep Learning Model Implementation

The EfficientNet-B0 model is built and trained in PyTorch with the following key configuration:

Hyperparameter	Value
Base architecture	EfficientNet-B0 (ImageNet pretrained)
Input image size	224 × 224
Batch size	16
Phase 1 epochs	15 (backbone frozen)
Phase 2 epochs	15 (full fine-tuning)
Initial learning rate	0.001
LR scheduler	ReduceLROnPlateau (factor = 0.3, patience = 3)
Loss function	Cross-Entropy Loss
Optimiser	Adam
Number of classes	2 (Real / Fake)
Dropout (head)	0.4, 0.3
Training device	CUDA (GPU T4)

The trained model weights and class metadata are saved to medicine_model_info.pth, which is loaded once at Flask application startup to minimise inference latency.

4.4 Web Application Implementation

The web application is built with Flask as the backend framework. Users access the application via a browser, select the detection module (expiry or fake/real), and upload a medicine image through a file input form. The frontend communicates with the Flask backend via

asynchronous Fetch API calls, receiving JSON responses containing detection results. The interface renders expiry status (manufacturing date, expiry date, days remaining) or classification prediction (Real / Fake with confidence score) without requiring a full page reload. File upload validation restricts accepted formats to JPEG and PNG to prevent malicious uploads.

4.5 Hardware and Software Requirements

Hardware:

- GPU: NVIDIA T4 (CUDA-enabled) for model training; CPU sufficient for inference
- RAM: Minimum 8 GB (16 GB recommended for training)
- Storage: SSD storage preferred for fast model loading

Software Stack:

- Python 3.x, PyTorch, torchvision, EfficientNet (timm)
- Tesseract OCR, pytesseract
- OpenCV, NumPy, Pillow
- Flask, HTML5, CSS3, JavaScript
- scikit-learn, matplotlib, seaborn (evaluation and visualisation)

5. Results and Discussion

5.1 Expiry Date Detection Performance

The OCR-based expiry date detection module was evaluated on a test set of medicine images captured under varying lighting and packaging conditions. The dual-pass Tesseract configuration (PSM 3 + PSM 6) demonstrated improved text coverage compared to single-pass OCR, particularly for labels with mixed text orientations. The regex engine successfully extracted dates in all five mandated CDSCO formats, with correct identification of both manufacturing (MFD) and expiry (EXP) date fields. Accuracy exceeded 95% for clearly printed labels, with performance degradation observed on heavily worn or low-contrast packaging, consistent with findings in literature on pharmaceutical OCR challenges.

5.2 Fake vs Real Classification Performance

The EfficientNet-B0 model, trained using the two-phase fine-tuning strategy, achieved a **validation accuracy of 99.78%** on the held-out validation set. The confusion matrix confirmed strong performance across both classes, with very few misclassifications. Performance metrics are summarised below:

Metric	Value
Training Accuracy (Phase 2)	>99%
Validation Accuracy	**99.78%**
Precision (Real class)	~99.8%

Metric	Value
Recall (Real class)	~99.8%
Precision (Fake class)	~99.7%
Recall (Fake class)	~99.7%

The two-phase training strategy proved critical: freezing the backbone during Phase 1 allowed the classification head to converge on domain-specific features without disrupting pretrained ImageNet representations, while Phase 2 fine-tuning allowed the full network to adapt to pharmaceutical visual patterns while avoiding catastrophic forgetting.

5.3 Comparative Analysis

Compared with conventional manual inspection and earlier handcrafted feature approaches, the proposed system offers significant advantages:

Method	Expiry Detection	Counterfeit Detection	Accessibility	Speed
Manual Inspection	Error-prone	Limited	High	Slow
Barcode/QR Scanning	Partial	No	Medium	Fast
Handcrafted Features (traditional ML)	N/A	~80–85%	Low	Medium
MediCheck (Proposed)	**>95%**	**99.78%**	**High (web-based)**	**Real-time**

5.4 Discussion

The results demonstrate that combining dual-pass Tesseract OCR with a carefully engineered regex pipeline is an effective strategy for robust expiry date extraction from Indian pharmaceutical packaging. The EfficientNet-B0 model with two-phase fine-tuning achieves near-perfect classification accuracy, confirming the suitability of lightweight pretrained CNNs for

pharmaceutical counterfeit detection. A key strength of the system is its accessibility: the Flask-based web interface requires no technical expertise from end users and supports operation on standard consumer devices including smartphones through a mobile browser.

One notable limitation is that OCR accuracy decreases on severely degraded or damaged labels, where preprocessing alone cannot recover lost textual information. For the fake-versus-real module, performance depends on the diversity and representativeness of the training dataset; the system may require retraining with additional samples when encountering novel packaging designs not present in the training data. Future work should address these limitations through expanded dataset curation and the integration of a fallback mechanism for low-confidence detections.

6. Conclusion

This paper presented MediCheck, an AI-powered web application that combines OCR-based expiry date detection and deep learning-based counterfeit medicine identification in a single accessible platform. The expiry detection module employs dual-pass Tesseract OCR with a layered regex engine aligned to CDSCO labelling standards, achieving over 95% detection accuracy. The counterfeit detection module fine-tunes an EfficientNet-B0 model using a two-phase transfer learning strategy, achieving a validation accuracy of 99.78%. Both modules are integrated into a responsive Flask web application enabling real-time, user-friendly medicine verification.

The proposed system offers practical value for pharmacies, hospitals, distribution centres, regulatory personnel, and individual patients — placing a reliable pharmaceutical verification tool directly in the hands of end users without requiring technical expertise. By automating two critical safety functions in one accessible platform, MediCheck contributes to reducing medication errors, protecting patient safety, and supporting pharmaceutical regulatory efforts aligned with CDSCO and WHO standards.

Future Enhancements:

- Mobile application deployment (Android/iOS) with native camera integration for real-time scanning
- Multi-language label processing to support regional Indian language pharmaceutical labels
- Cloud deployment (AWS/Google Cloud) for scalable hospital and pharmacy access
- Real-time camera feed support eliminating the need for manual image upload
- Integration with hospital inventory management and CDSCO regulatory systems
- Admin dashboard for detection history tracking and database management

Acknowledgment

The authors express sincere gratitude to Sri Raghu Kalidindi, Chairman, and Dr. Ch. Srinivasu, Principal, Raghu Engineering College, for providing the necessary facilities. The authors thank Dr. K. V. Satyanarayana, Program Head, Department of CSE (Data Science), and Mr. M. V. Ramana Murthy, Assistant Professor, for guidance and supervision throughout this work. The authors also extend thanks to all faculty members and non-teaching staff of the Department of Computer

Science and Engineering, Raghu Engineering College, Dakamarri, Visakhapatnam.

References

1. Smith, R. (2007). An overview of the Tesseract OCR engine. *Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR)*, 2, 629–633. IEEE.
2. Chiticariu, L., Li, Y., & Reiss, F. R. (2013). Rule-based information extraction is dead! Long live rule-based information extraction systems. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 827–832.
3. World Health Organization. (2017). *A study on the public health and socioeconomic impact of substandard and falsified medical products*. WHO Press, Geneva.
4. World Health Organization. (2021). *Substandard and falsified medical products*. Retrieved from https://www.who.int/news-room/fact-sheets/detail/substandard-and-falsified-medical-products
5. Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *Proceedings of the 36th International Conference on Machine Learning (ICML)*, PMLR 97, 6105–6114.
6. Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., & Liang, J. (2016). Convolutional neural networks for medical image analysis: Full training or fine tuning? *IEEE Transactions on Medical Imaging*, 35(5), 1299–1312.
7. Ketkar, N., & Santana, E. (2021). *Deep Learning with Python* (2nd ed.). Apress. Chapter 12: Deploying Deep Learning Models.
8. Karatzas, D., Gomez-Bigorda, L., Nicolaou, A., Ghosh, S., Bagdanov, A., Iwamura, M., & Llorens, D. (2015). ICDAR 2015 competition on robust reading. *Proceedings of the 2015 International Conference on Document Analysis and Recognition (ICDAR)*, 1156–1160.
9. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.
10. Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
11. LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444.
12. Vijayalakshmi, B., & Mohan, V. (2019). Skin lesion classification using deep learning with EfficientNet. *Biomedical Signal Processing and Control*, 52, 128–137.
13. Kavin, R., & Shirley, C. (2024). OCR-based extraction of expiry dates and batch numbers in medicine packaging for error-free data entry. *Semantic Scholar*. Retrieved from https://www.semanticscholar.org

14. International Journal of Engineering Research and Technology (IJERT). (2022). Counterfeit medicine detection using deep learning. Retrieved from https://ijirt.org

15. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., & Chintala, S. (2019). PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32.

16. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., & Zheng, X. (2016). TensorFlow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI)*.

17. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.

18. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779–788.

19. National Pharmaceutical Pricing Authority (NPPA). (2021). *Guidelines on pharmaceutical labelling*. Retrieved from https://www.nppa.gov.in

20. Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*.