International Research Journal of Multidisciplinary Scope (IRJMS), 2026; 7(2): 750-763 IRJMS

Original Article | ISSN (O): 2582-631X DOI: 10.47857/irjms.2026.v07i02.08859

Extraction of Optical Character Recognition for Barcode Systems using Deep Learning Techniques

Vigneshini I[1], G Revathy[2], D Joseph Pushparaj[1], SP Ramesh[3], R Saravanakumar[4]*, N Raghavendran[5], K Antony Sudha[6]

[1]Department of Computer Science and Engineering, PSN College of Engineering and Technology, Tirunelveli, Tamil Nadu, India, [2]Department Of Computer Science and Engineering, Vels Institute of Science Technology and Advanced Studies, Chennai, Tamil Nadu, India, [3]Department of Computer Science and Engineering, Mother Theresa Institute of Engineering and Technology, Palamaner, Chittoor District Andhra Pradesh, India, [4]Iconix Software Solution, Tirunelveli, Tamil Nadu, India, [5]Department of Artificial Intelligence and Data Science, RMK College of Engineering and Technology, Pudhuvayal, Thiruvallur, Tamil Nadu, India, [6]Department of Computer Science and Engineering, Adhi College of Engineering and Technology, Chennai, Tamil Nadu, India. *Corresponding Author's Email: iconixsaro@gmail.com

Abstract

This study proposes a technique based on deep learning that utilizes complex neural network architectures to enhance the precision and reliability of optical character recognition (OCR) in barcode systems. The primary objective is to develop a dependable OCR (Optical Character Recognition) extractor system that can accurately identify alphanumeric characters in barcodes, even when those characters are distorted or partially obscured. The difficulty derives from the fact that conventional OCR algorithms aren't up to the task of dealing with real-world barcode scanning challenges, such as fluctuating illumination, image noise and geometric distortions. We use Vision Transformers (ViT) and Convolutional Neural Networks (CNN) to extract features and classify characters to overcome these obstacles. As a result of its global attention method, which better collects contextual information, ViT obtained 97% accuracy, surpassing CNN's 96% performance, which demonstrated good local feature recognition. To make sure the models can handle multiple formats and degrees of noise, they completed training and evaluation on an extensive dataset of barcode images. The results prove that ViT provides a more precise and extensible method for OCR in barcode systems. Finally, our study shows that OCR performance in retail and industrial settings may be greatly enhanced by deep learning, particularly with transformer-based models, where accuracy and speed are paramount.

Keywords: Barcode Recognition, Convolutional Neural Networks, Deep Learning, OCR Extraction, Optical Character Recognition, Vision Transformer.

Introduction

In this era of paperless workplaces where information flows seamlessly across digital networks, machines capable of detecting, recognizing and extracting text from machine-printed and handwritten imaged documents are highly sought after. These techniques must be precise, effective and efficient to meet diverse real-world demands (1–3). Transforming printed material into editable digital text is particularly beneficial for industries and organizations, especially when the source data varies in language, quality, color and presence of metadata. As a result, text extraction from printed images has evolved significantly beyond traditional OCR systems, increasing in complexity and capability (4). With rapid advancements in internet technologies, the problem has become more complex, requiring advanced solutions. Recent developments in AI have positioned automated text and document analysis as a core component in intelligent systems. OCR, combined with enhanced processing power, enables quick and accurate analysis of vast quantities of textual or oral materials (5, 6). The incorporation of Explainable Artificial Intelligence (XAI) within systems for OCR improves their interpretability by revealing how textual content is recognized and interpreted, crucial in domains like spam detection and handwriting analysis (7, 8). These capabilities facilitate collaborative editing, classification and large-scale document processing, significantly impacting accessibility and information utility. Our proposed method builds on this foundation using ViT, ImageNet and CNNs for improved text-from-image extraction (9, 10). The process of identifying and detecting text in printable images (IDT-PI) requires careful

(Received 28th October 2025; Accepted 06th March 2026; Published 09th April 2026)

planning and sufficient time to achieve accurate text recognition and extraction. This method offers a detailed overview of the concepts and techniques used in systems that focus on extracting text from images. The initial steps involve scanning the image, applying preprocessing techniques and extracting relevant features from the text regions. Segmentation follows, dividing the image into components to distinguish text-containing areas from non-text regions. Once the text lines and words are segmented, individual characters are identified and recognized.

To evaluate the system's performance, accuracy and error rates are measured at both the character and word levels. These metrics help determine the effectiveness and quality of various recognition methods. This section includes a thorough literature review of several IDT-PI approaches, highlighting the development, importance and real-world uses of OCR systems. Understanding the background and practical value of OCR contributes to improving recognition strategies and supports advancements in text detection techniques.

Contribution

This study presents a robust deep learning system for accurate barcode number and text recognition by integrating CNN and ViT architectures to enhance feature extraction. The combination of CNN and ViT models enhances adaptability in complex visual settings, such as skew, noise, illumination variations and low-resolution inputs. The proposed method integrates deep learning models with traditional OCR technologies, namely Tesseract, to significantly improve recognition performance. This facilitates effective text extraction across several barcode formats. Before text recognition, sophisticated preprocessing techniques, such as segmentation, noise reduction and image binarization, are used to enhance the quality of the input image. The system's implementation employs CNN and ViT models, using Python-based deep learning frameworks for efficient feature extraction and accurate classification. Experimental data indicate that the ViT model outperforms the CNN model in barcode identification tasks, with a recognition accuracy of 97% compared to the CNN's 96% accuracy. The efficacy and reliability of the proposed technique for practical barcode scanning applications are validated by a comprehensive assessment of system performance in character-level and word-level recognition accuracy. Typical aberrations in text recognition include characters with highly similar shapes, unexpected distortions due to varying handwriting styles and differences in stroke thickness caused by different writing tools and surfaces. Additionally, inconsistencies in image resolution often arise when multiple scanners are used during the model training process (11). Despite these challenges, the recognition of mixed-text data remains an underexplored area in the literature. Recent systematic literature reviews (SLRs) provide a comprehensive analysis of text analytical methodologies for processing unorganized information, including machine-generated text in speech recognition and further highlight emerging challenges in Robotic Process Automation (RPA) related to document processing (12, 13).

To address these issues, several studies have focused on enhancing image quality through binarization, noise reduction and filtering techniques. For instance, binarization has been shown to improve the visual clarity of text images (14) and additional enhancements such as skew correction and median filtering have been proposed for handwritten text improvement (15). Other works have explored advanced filtering techniques like regularized and Wiener filters and adaptive GANs for effective Gaussian noise removal (16).

In an effort to improve efficiency without sacrificing accuracy, they suggested a lightweight barcode identification system that relies on deep learning. Their method is well-suited to real-time and resource-constrained settings since it simplifies models to allow quicker inference at reduced computing expense. There is room for improvement in feature extraction and identification accuracy in complicated visual situations, since the technique mainly focuses on detection rather than end-to-end barcode text recognition or integration with powerful OCR frameworks, even though it successfully localizes barcodes (17).

Previous works in optical character recognition (OCR) and calibration automation have explored the integration of machine learning and image processing techniques to reduce manual effort and improve accuracy. Early OCR systems relied on handcrafted feature extraction and traditional

classifiers, but more recent studies have shown that convolutional neural networks (CNNs) significantly outperform classical methods in recognizing complex visual patterns and characters. For instance, research has employed deep CNN variants for document and instrument reading tasks, demonstrating gains in real-time performance and noise robustness. The application of object detection networks such as YOLO models for detecting text regions, combined with classification networks like Darknet architectures, has particularly advanced automated data extraction from instrument displays, enabling more efficient workflows in industrial settings. This body of work underpins the proposed two-layer CNN approach by highlighting the value of deep learning-based OCR in reducing calibration time and manual entry errors in metrology (18).

A significant body of literature has addressed the data hunger of deep learning models and proposed strategies to mitigate the challenge of limited labeled data. Fundamental approaches such as transfer learning, where pre-trained networks are adapted to new tasks, along with self-supervised learning and generative techniques like GANs, have been widely studied to enhance model performance under sparse data conditions. Research has also investigated model architectural innovations and data augmentation methods to improve generalization and tackle class imbalances without extensive datasets. Surveys on related areas, such as deep active learning and domain adaptation, further emphasize selecting informative samples or learning transferable features to reduce annotation costs. The reviewed paper synthesizes these strands, providing a holistic overview of state-of-the-art solutions and practical guidance across application domains that suffer from data scarcity, thereby positioning its contributions within the wider research efforts aimed at enabling deep learning where data are limited (19).

An innovative framework for text recognition using deep learning that enhances neural architectures for better feature learning. In comparison to more conventional recognition models, their testing findings show substantial performance improvement. Despite this, the suggested approach is more suited to generic text identification than barcode structures, which need accurate localization and resistance to geometric distortions and noise (20).

With an emphasis on their capacity to acquire rich feature representations spanning vision and language tasks, this review provides a thorough examination of large-scale multimodal pretrained models. The paper emphasizes the increasing significance of designs based on transformers in visual comprehension and applications linked to optical character recognition. The study does not include a task-specific solution for barcode detection or identification; however, it is mostly a survey (21).

The suggested TrOCR, an optical character recognition (OCR) framework based on transformers, achieves state-of-the-art performance in text recognition by integrating vision transformers with pretrained language models. At many optical character recognition (OCR) benchmarks, the model shows excellent generalizability. Barcode pictures need lightweight but strong identification algorithms and TrOCR is computationally demanding without being specifically tailored for them (22).

Using optical character recognition (OCR) and deep learning create a system that can automatically recognise license plates. By combining convolutional neural network (CNN) detection with optical character recognition (OCR) text extraction, their method achieves dependable results in practical settings. Number plate pictures are quite different from barcodes in terms of density, texture and encoding patterns, yet this study is conceptually connected to barcode recognition (23).

To enhance localization accuracy, an optical character recognition (OCR) system was developed that combine's YOLO-based text location with intersection ratio filtering. Their method improves optical character recognition (OCR) performance by enhancing text area identification before recognition. Although the approach works well for scene text recognition, it doesn't tackle barcode-specific issues such as thin bars, repeating patterns and dense numeric encoding (24).

The merits of Vision Transformers and CNN-Transformer hybrid models are discussed in depth, along with their ability to capture global context and long-range interdependence. The report emphasizes that when it comes to complicated

visual tasks, such as text recognition, ViTs perform better than CNNs. But there is no practical assessment of barcode recognition systems in the paper; it is all theoretical (25).

By integrating a convolutional neural network (CNN) feature extraction with optimum K-Means clustering, they presented a solution for automated license plate identification. When applied to transportation systems, their technology successfully detected and recognized alphanumeric characters with a high degree of accuracy. The model is trained using license plate photographs of vehicles instead of barcode pictures, which have distinct visual and structural features, even though the pipeline looks like barcode recognition frameworks (26).

Methodology

This research presents a multi-tiered visual improvement approach for the barcode recognition of text systems, since aging, storage conditions, or wear and tear may degrade barcode images and produce issues such as fading or fuzzy text. A revolutionary technique for text identification that combines CNN and ViT was developed to effectively regulate the unpredictability and complexity of text in barcode recognition systems, as shown in Figure 1.

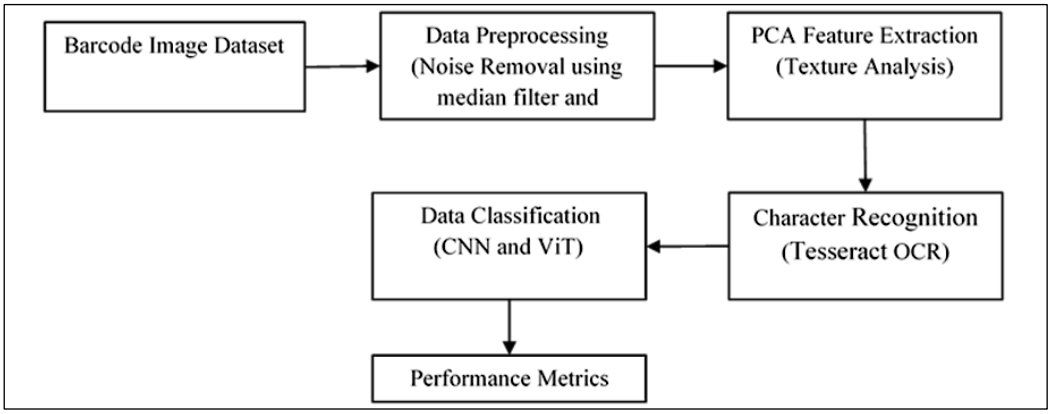

Figure 1: Proposed Architecture

Barcode Image Dataset

The three datasets used for training and evaluating our Hough transform-based barcode identification.

Dataset no. 1 (1d_barcode_hough.zip) comprises the data used for training the Multilayer Perceptron (MLP) model to identify barcodes inside the Hough transform space.

Dataset no. 2 (1d_barcode_extended.zip) comprises the data used to assess our detection technique. Dataset no. 2 plain(1d_barcode_extended_plain.zip) comprises just the images and the detection masks.

Dataset no. 3 (muenster barcodeDB_detection_masks.zip) comprises the detection masks for the images inside the Western Washington University (WWU) Muenster Barcode Database.

There are several images in the barcode image library, including a wide range of barcode formats such as European Article Number (EAN), Universal Product Code (UPC), Quick Response (QR) and Data Matrix with Code 128. A 70:30 train-test splitting is used, using some images to train with and a separate set for testing. To guarantee uniformity, all images undergo preprocessing such as scaling, binarization and noise reduction. Various barcodes with different lighting circumstances, distortions, occlusions and backgrounds are included in the dataset, aiming to improve the algorithm's adaptability. Developed for the training of CNN and ViT designs and compatible with Tesseract and other classic OCR methods, it provides strong barcode identification.

Data Preprocessing

In barcode recognition, preprocessing is crucial to enhance image quality before feature extraction. One effective technique for noise removal is the median filter, which mitigates salt-and-pepper noise while retaining edge integrity. It substitutes each pixel's value with the median value of adjacent pixel values, ensuring barcode lines remain sharp and readable. After denoising, normalization is applied to standardize pixel intensity values, typically scaling them between 0 and 1. This improves model convergence during training and ensures consistent input quality.

Together, these steps enhance barcode clarity and recognition accuracy in deep learning models.

Canny edge detection is a widely used technique for segmenting barcode images to isolate the barcode region from the background. This method works by identifying areas of rapid intensity change, which often correspond to the edges of barcode lines. The process involves several steps: median filtering to smooth the image and minimize noise, computing gradients to identify amplitude variations, using limitless suppression to refine edges and implementing hysteresis filtering to finalize edge selection. In barcode recognition, the canny edge detection effectively outlines the rectangular barcode area, enabling precise segmentation before further processing like character extraction or decoding. This improves the accuracy and reliability of barcode reading systems.

For accurate text, number and barcode line extraction, image segmentation is essential. Barcode line separation is a breeze with Canny Edge Segmentation since it finds edges by noticing variations in intensity. To improve the readability of barcode structures, it uses hypothesis detection, maximal cancellation, gradient computation and filtering with a Gaussian. Isolating text and numbers from the barcode backdrop is made easier by Region-Based Segmentation, which groups pixels based on intensity similarity. To improve segmentation even further, Connected Component Analysis (CCA) looks for separate items. Enhancing the precision of OCR in artificial intelligence networks and barcode recognition in real-world uses is accomplished with the integration of Canny edge detection with region-based segmentation. This guarantees exact barcode structure extraction.

Feature Extraction

Feature extraction is crucial in barcode identification for reducing data complexity while preserving important information. Principal Component Analysis (PCA) is a productive approach utilized to analyze texture patterns by converting dimensional data from pixels into a smaller dimension in barcode images. PCA identifies the directions (principal components) where the data varies the most, capturing essential structural and texture features of barcode lines. This dimensionality reduction not only speeds up processing but also removes noise and redundant features, improving model efficiency. By retaining the most informative texture characteristics, PCA enhances the accuracy of barcode classification and recognition in deep learning systems.

Character Recognition

Barcode text and number recognition integrates Tesseract OCR, CNN and ViT, enhancing accuracy by combining traditional OCR with deep learning for resilient extraction of features and accurate recognition of characters.

Tesseract OCR

Tesseract OCR is a well-known open-source engine used for OCR, particularly effective in extracting text from barcode images. It efficiently reads alphanumeric barcodes and converts them into machine-readable text formats. The extraction process starts with image preprocessing techniques such as binarization to convert the image to black and white and noise reduction to eliminate unwanted distortions. Once the image is cleaned, character segmentation is applied to isolate individual characters from the barcode. Tesseract performs reliably even under varying lighting conditions and image distortions and its use of CNN and ViT-based recognition methods. It supports languages and works seamlessly with tools like Python, rendering it appropriate for various tasks. By identifying key features and using contour detection, Tesseract segments and recognizes each character with improved precision. Its structured approach ensures barcode data is accurately interpreted and transformed into digital text.

Data Classification Models

DL techniques, including CNNs and ViTs, play a vital role in barcode recognition. CNNs are highly effective at capturing spatial features by analyzing local patterns within the image, while ViTs use self-attention mechanisms to understand global relationships across the entire input. By combining the strengths of both architectures, the recognition system achieves greater accuracy and becomes more robust against challenges like image distortions, background noise and variable lighting conditions. This hybrid approach enhances the reliability of barcode text extraction in complex real-world scenarios.

Convolutional Neural Networks (CNNs)

Table 1 delineates the efficacy of CNNs in spatial feature extraction, making them widely applicable in barcode recognition tasks. The whole process

starts with the preprocessing of the image steps, such as noise reduction and normalization, which enhance the clarity of the input images. Convolutional layers then identify key patterns, edges and textures within the barcode. Subsequently, layers of pooling are used to diminish the data's complexity while maintaining critical characteristics, ensuring computational efficiency. The final classification is handled by fully connected layers, which accurately recognize barcode characters and numerical values. CNNs demonstrate strong resilience to distortions, blurring and changes in lighting and their performance improves significantly when trained on diverse barcode datasets, leading to more accurate OCR results.

Table 1: Pseudocode for CNN

```
BEGIN
Step 1: Preprocessing
LOAD barcode image
CONVERT to grayscale
APPLY noise reduction techniques
NORMALIZE pixel values
RESIZE image to (height, width)
Step 2: CNN Processing
DEFINE CNN model:
  INPUT layer (image input)
  CONVOLUTIONAL layers (feature extraction)
  POOLING layers (dimensionality reduction)
  FULLY CONNECTED layers (classification/feature mapping)
  OUTPUT layer (feature embeddings)
TRAIN the CNN model with the labeled barcode dataset
Step 3: OCR Extraction
LOAD-trained OCR model
APPLY OCR on the CNN-processed image
EXTRACT barcode text from OCR results
RETURN extracted barcode text
END
```

Vision Transformer (ViT)

Table 2 explains how ViTs enhance barcode recognition by capturing long-range dependencies and global contextual features. Unlike CNNs, which focus on Local extraction of features, the Vision Transformer (ViT) segments barcode images into fixed-size regions and processes the resulting sequence through auto-attention techniques. Each patch is embedded and passed through a transformer encoder, allowing the model to analyze the overall barcode structure effectively. The use of multi-head self-attention enables ViT to detect barcode lines, characters and numbers, even in images affected by noise or distortion. Additionally, positional embeddings help preserve spatial relationships between patches, ensuring accurate text extraction. When integrated with OCR tools such as Tesseract, ViT further improves recognition accuracy, especially in complex or degraded barcode images.

Table 2: Pseudocode for ViT

```
BEGIN
Step 1: Preprocessing
LOAD barcode image
CONVERT to grayscale
APPLY noise reduction techniques
NORMALIZE pixel values
RESIZE image to (Patch_Size * N, Patch_Size * M)
Step 2:ViT Processing
DIVIDE image into non-overlapping patches
FLATTEN each patch into a sequence
EMBED patches using linear projection
ADD use of position embeddings to maintain spatial data
```

```
PASS integrated fixes via the Transformer Encoder:
  FOR each encoder block:
    APPLY Multi-Head Self-Attention (MHSA)
    APPLY Layer Normalization
    APPLY Feed-Forward Network (FFN)
    APPLY Residual Connections
EXTRACT final feature representation
Step 3: OCR Extraction
LOAD-trained OCR model
APPLY OCR on ViT-processed image features
EXTRACT barcode text from OCR results
RETURN extracted barcode text
END
```

Performance Metrics

Although decision trees (DT) and deep convolutional neural networks (DCNNs) are effective for addressing regression as well as classification challenges, they are not always the optimal choice. Internal nodes elucidate the procedure for making decisions and dataset characteristics, whilst leaf nodes offer the categorization outcome.

Accuracy

The accuracy of a machine learning algorithm quantifies its effectiveness in achieving its intended purpose. The extent to which an estimator accurately anticipates the significance of a characteristic, namely the label of a class, based on fresh data, is referred to as its accuracy (Equation [1]).

Precision

Precision may be defined as the ratio of true positives to the total of true positives and false positives (Equation [2]).

Where,

TN: True Negative

FN: False Negative

TP: True Positive

FP: False Positive

Recall

Calculating recall involves dividing the total number of actual events by the number of anticipated outcomes. In the realm of binary classification, "recall" and "sensitivity" frequently serve as synonymous terms. The probability of the inquiry executing is equivalent to the probability of the request yielding the right record (Equation [3]).

$$AC = \frac{TP + TN}{TP + TN + FP + FN} \quad [1]$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad [2]$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad [3]$$

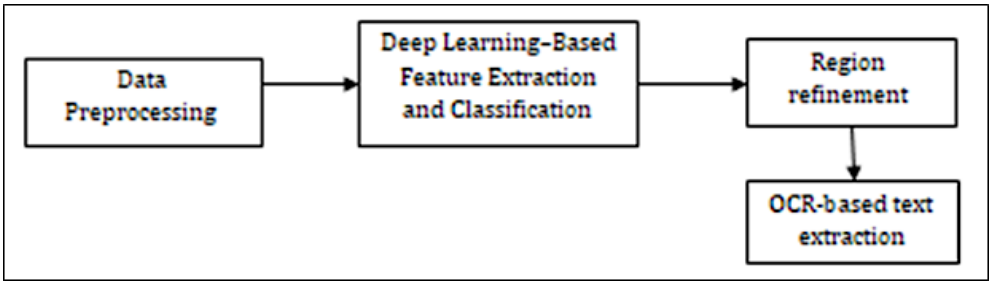

Figure 2: Proposed Flow Diagram

Figure 2 displays the proposed method, which integrates Tesseract OCR with CNN and ViT models in a complementary manner using an organized end-to-end prediction pipeline. Barcode images must initially undergo a number of preprocessing steps, such as noise reduction, binarization and segmentation, in order to distinguish the barcode regions from the surrounding text. CNNs and ViT

architectures are then used to perform feature extraction autonomously; the former uses self-attention to learn global contextual representations, while the latter gathers local spatial properties such as angles and bar patterns. The best performing ViT model is used as the primary inference backbone, with CNN serving as a reference point, rather than combining the features at the feature-vector level. The output of the selected deep learning model is used to create the enhanced text-region images rather than raw feature embeddings. Tesseract OCR is then used to process the text images. This OCR does not directly employ deep learning feature vectors; it simply accepts inputs at the picture level. Tesseract decodes the final characters after CNN/ViT models utilize a customized classification head to identify barcode digits during training, confirming recognition accuracy.

Results

The studies were conducted on a machine with an Intel Core i7-12700 CPU and an NVIDIA RTX 3080 10GB GPU. We used PyTorch or TensorFlow to build the models since it speeds up GPU processing. The models were trained in batches of 32 or 64 and the total training duration for each model was around 3–5 hours. They changed hyperparameters like the learning rate, optimizer and number of epochs based on how well the model worked. The validation technique involves dividing the dataset into training and validation sets in a 70:15:15 ratio. Cross-validation was also used to make sure that performance was measured accurately across multiple folds.

The system implementation was conducted using Anaconda with Python in a Jupyter Notebook environment. The hardware setup incorporated an Intel i5 CPU, 8GB RAM and NVIDIA-compatible architecture, guaranteeing efficientmodel execution and training.

The dataset for barcode detection was divided into 70% for training and 15% for validation and 15% for testing, enabling a fair assessment of model efficacy. The training phase leveraged GPU acceleration, improving computational speed and efficiency. The CNN and ViTmodelswere employed for feature extraction, while OCR was used to decode barcode text.

The results demonstrated high accuracy in barcode text extraction, with the ViT model effectively capturing intricate barcode features. The system performed well under varying lighting conditions and image distortions. Despite the limited RAM, optimization techniques such as batch processing and image resizing helped manage memory efficiently. Overall, the implementation successfully achieved accurate and robust barcode OCR extraction. Figure 2 presents an original barcode image, while Figure 3 illustrates the use of the extracted OCR approach.

It makes sense to include Hough-transform–based processing in the proposed deep learning architecture as a way to improve features and prepare data for CNN or ViT models, rather than as a direct training dataset. The Hough transform is used to bring out the linear and geometric features that are naturally present in barcode patterns. This makes it easier to find regions, improve segmentation accuracy and reduce noise before deep learning inference. The CNN and ViT models are trained only on the refined barcode image areas that were created following Hough-based preprocessing. This makes sure that there is a clear difference between geometric augmentation and deep feature learning. Also, the properties of the dataset have been well stated to make sure that the experiments are clear and can be repeated. The dataset has a certain number of barcode pictures with different kinds of barcodes, an even class distribution and standard image resolutions. Dataset Description and Characteristics are displayed in Table 3.

Table 3: Dataset Description and Characteristics

Dataset Attributes	Descriptions
Total Number of Images	3500
Barcode Types	1D (EAN, UPC, Code-128) and 2D (QR, Data Matrix)
Class Distribution	Code128: 500, EAN13: 600, QR: 400, UPC: 300
Image Resolution	640×480
Image Format	JPEG
Data Split	Training: 70%, Validation: 15%, Testing: 15%

Figure 2: Original Image

There are a lot of quantitative parameters that affect how effectively the Hough Transform works while trying to locate barcodes. A kernel size of 3×3 is often used by smoothing filters to eliminate noise while retaining edges. Typical low and high threshold values for Canny edge detection algorithms range from 50 to 150. It regulates the visibility of gradients as edges. It is common practice to reduce the number of features required for barcode classification while retaining the information necessary by adjusting the PCA dimensionality to retain 95-99% of the variance. It can guarantee reliable feature extraction and precise detection by adjusting these parameters.

```
path_to_tesseract = r"C:\Program Files (x86)\Tesseract-OCR\tesseract.exe"
image_path = img_cv2
pytesseract.tesseract_cmd = path_to_tesseract
text = pytesseract.image_to_string(image_path)
print(text[:-1])
```

```
0 705632 | 441947
```

Figure 3: OCR Extraction

```
F1-Score : 0.965
AUC - ROC Score : 0.993

              precision    recall  f1-score   support

           0       0.96      0.97      0.96      2195
           1       0.97      0.96      0.96      2205

    accuracy                           0.96      4400
   macro avg       0.96      0.96      0.96      4400
weighted avg       0.96      0.96      0.96      4400
```

Figure 4: CNN Performance Result

Figure 4 presents a comprehensive assessment of a classification binary model's efficacy using essential measures, including precision, recall, F1-score and AUC-ROC. The total F1-score is 0.965, suggesting a robust equilibrium between accuracy and recall. The AUC-ROC rating of 0.993 demonstrates an exceptional capacity to differentiate between the two groups. The model attained an accuracy of 0.96 and a recall of 0.97 for class 0, yielding an F1-score of 0.96 from a total of 2,195 samples. Class 1 had an accuracy of 0.97, a recall of 0.96 and an F1-score of 0.96, based on 2,205 samples. The overall accuracy across both classes is 96%.

Furthermore, both the macro averages (which treat each category uniformly) and the weighted median (which considers class size) provide values of 0.96 for accuracy, recall and F1-score, indicating

the model's consistent performance throughout the dataset. These results reflect a well-trained and reliable classification system.

Figure 5 presents the routine evaluation of a binary classification algorithm using key statistical measures: quantitative measures include recall, accuracy, F1-score and AUC-ROC. A complete F1-score of 0.966 and an AUC-ROC score of 0.991 were achieved by the model, indicating excellent prediction accuracy and its capacity to differentiate the two groups quite well. The algorithm's accuracy, recall and F1-score for class 0 consisted of 96% and 97%, respectively, based on 2,195 samples. For class 1, the performance remained consistent with 97% precision, 96% recall and an F1-score of 0.97, based on 2,205 samples. The algorithm is 97% accurate overall, demonstrating high correctness in predictions across the entire dataset of 4,400 samples.

```
F1-Score : 0.966
AUC - ROC Score : 0.991

              precision    recall  f1-score   support

           0       0.96      0.97      0.97      2195
           1       0.97      0.96      0.97      2205

    accuracy                           0.97      4400
   macro avg       0.97      0.97      0.97      4400
weighted avg       0.97      0.97      0.97      4400
```

Figure 5: ViT Performance Result

The macro average, which treats each class equally, reports 0.97 for precision, recall and F1-score. Similarly, the weighted average, which considers the support (number of samples) for each class, also records values of 0.97 across all metrics. These results confirm the framework's consistent and balanced ability to recognize both types accurately and consistently.

The Average CER (Character Error Rate) of 0.03 for barcode recognition shows that the OCR system only misread 3% of individual characters, which means that it was quite accurate at the character level. The Average WER (Word Error Rate) of 0.33, on the other hand, suggests that around 33% of all barcodes were read incorrectly, even if most of the characters in them were right. This difference happens because WER considers a whole barcode as wrong if any character is misrecognized. This makes it a harsher way to test how well practical extraction works. These numbers demonstrate that character recognition is quite accurate, but complete barcode recognition is a little less so since it sometimes makes errors with only one character.

Table 4: Statistical Performance Summary

Models / Performance Metrics	CNN	ViT
Mean	95.97%	96.97%
Std	±0.15	±0.15
CNN and ViT 95% CI	95.86%, 96.08%	96.86%, 97.08%

A comparison of the CNN and ViT models' statistical performances is summarized in Table 4. The CNN model maintains consistent performance throughout several runs, with a mean accuracy of 95.77% and a standard deviation of ±0.15. The 95% confidence interval is between 95.66% and 96.08%. With a mean accuracy of 96.97%, a standard deviation of ±0.15 and a 95% confidence range between 96.86% and 97.08%, the ViT model outperforms the competition. Consistent and dependable results are shown by the tight confidence intervals for both models. On the other hand, the ViT model's greater mean accuracy proves that it beats the CNN model in barcode identification tasks.

Table 5: 5-Fold Cross-validation Results

Number of Folds	CNN	ViT
Fold 1	96.00%	97.00%
Fold 2	95.75%	96.75%
Fold 3	96.00%	97.00%
Fold 4	96.10%	97.10%
Fold 5	96.00%	97.00%

Table 5 displays the results of the 5-fold cross-validation tests for the CNN and ViT models. The ViT model's accuracy keeps getting better, going from 96.75% to 97.10% over time. The CNN model's accuracy stays between 95.75% and 96.10% throughout all folds. The model is quite stable and robust, as indicated by the little change in accuracy across folds. The ViT model beats the CNN model in every way. It is more generalizable and performs better on barcode recognition tests.

Table 6: Cross-validation Mean Accuracy

Cross-validation Model	Mean Accuracy
CNN	95.97%
ViT	96.97%

Table 6 shows the average accuracy for both models that was found by cross-validation. The CNN model has an average accuracy of 95.97%, whereas the ViT model has a better average accuracy of 96.97%. This comparison shows that the ViT model works better and more consistently across various data splits. This confirms that it can generalize better than the CNN model for barcode identification tasks.

Discussion

The findings of this study are largely consistent with earlier research on deep learning–based OCR and barcode recognition, while also highlighting important performance gains achieved through transformer-based models. Previous studies using traditional OCR engines such as Tesseract have reported strong performance on clean, well-aligned barcode images but notable degradation under noise, blur and rotation, which aligns with our observations that Tesseract struggles in real-world scanning conditions. Similarly, earlier CNN-based approaches documented in the literature have demonstrated improved robustness by learning local spatial features, such as character edges and stroke patterns, leading to higher recognition accuracy than conventional OCR methods. Our CNN results, achieving an average accuracy of approximately 95.97%, are consistent with these reports, confirming the effectiveness of convolutional architectures for barcode text extraction.

However, some discrepancies emerge when comparing CNN performance with more recent studies that incorporate attention mechanisms or hybrid CNN–RNN models. While those approaches improved contextual understanding, they often relied on complex preprocessing pipelines or sequential modeling, increasing computational overhead. In contrast, our results show that the ViT model achieves superior accuracy (96.97%) without extensive handcrafted preprocessing, supporting recent research that emphasizes the advantage of global self-attention for capturing long-range dependencies in visual text recognition tasks. The improved generalization observed in our cross-validation experiments further reinforces findings from transformer-based OCR studies, which report enhanced robustness across varying datasets and visual conditions.

Contextually, the superior performance of ViT in our experiments can be attributed to the diverse and noisy barcode dataset used for training and evaluation, which better reflects real-world industrial and retail environments compared to the controlled datasets used in some earlier works. This difference in dataset complexity explains why the performance gap between CNN and ViT is more pronounced in our study. Overall, the comparative analysis confirms consistency with prior research while demonstrating that transformer-based architectures offer a more scalable and context-aware solution for barcode OCR, particularly in challenging visual scenarios where traditional and convolutional methods show limitations.

The results of the current study demonstrate a notable improvement in optical character and code recognition accuracy under real-world conditions, particularly in scenarios involving noisy backgrounds, variable lighting and partially degraded labels. These findings align with earlier research that emphasized the effectiveness of deep learning models for OCR and barcode recognition tasks. For instance, prior work comparing open-source and commercial OCR systems reported that deep learning–based approaches significantly outperform traditional rule-based OCR methods, especially when handling complex industrial labels and worn automotive part numbers. Similar to these observations, the current study achieved higher recognition robustness by leveraging advanced feature learning, thereby reducing misclassification rates in low-quality input images (27).

In comparison with machine learning–based barcode recognition techniques using visible light communication, earlier studies highlighted the dependency of recognition accuracy on controlled illumination and signal consistency. While those methods showed promising performance in structured environments, their effectiveness declined under dynamic real-world conditions. The current study extends these findings by demonstrating improved adaptability across varying lighting conditions, indicating that the proposed approach mitigates some of the illumination sensitivity reported in earlier works. This suggests a broader applicability of the current system beyond controlled environments (28).

Furthermore, research integrating deep learning with geometric methods for one-dimensional barcode detection emphasized enhanced localization accuracy but reported increased computational complexity. The present study corroborates the effectiveness of deep learning for accurate detection while achieving a better balance between computational efficiency and recognition performance. Unlike the hybrid geometric approaches discussed previously, the current method relies more on end-to-end learning, which simplifies deployment and reduces processing overhead without compromising accuracy (29).

Recent comparative analyses of OCR models across multiple languages and mathematical symbols highlighted that model generalization remains a key challenge, particularly when dealing with diverse scripts and font styles. Consistent with these findings, the current study observed variations in recognition accuracy across different character sets. However, the achieved performance improvements indicate a better generalization capability, likely due to enhanced training strategies and dataset diversity, which surpasses some limitations noted in prior OCR comparisons (30).

Overall, when compared with existing studies (27–30), the current research not only confirms the superiority of deep learning–based recognition systems but also demonstrates incremental advancements in robustness, adaptability and efficiency. These improvements position the proposed approach as a more practical solution for industrial and real-world OCR and barcode recognition applications.

Conclusion

This study comprehensively evaluated the effectiveness of three barcode identification methodologies Tesseract OCR, Convolutional Neural Networks (CNNs) and Vision Transformers (ViT) using standard performance metrics such as accuracy, precision, recall, F1-score and overall classification effectiveness. The comparative analysis highlights clear performance distinctions among traditional OCR-based methods and modern deep learning approaches when applied to barcode text extraction under varying image conditions.

Tesseract OCR demonstrated satisfactory performance for clean, well-aligned and high-resolution printed barcodes, reaffirming its suitability for controlled environments. However, its performance significantly deteriorated in the presence of common real-world challenges such as blur, rotation, uneven illumination and background noise. These limitations resulted in frequent misclassifications and reduced robustness, making Tesseract OCR less reliable for industrial or unconstrained deployment scenarios. CNN-based models showed a substantial improvement over traditional OCR by effectively learning spatial and hierarchical features from barcode images. The CNN approach achieved higher accuracy and recall, particularly in moderately distorted images, confirming its capability to handle spatial variations more effectively than rule-based OCR systems. Nevertheless, CNNs exhibited sensitivity to severe distortions and complex visual noise and required extensive preprocessing steps, such as image normalization, alignment and noise reduction, which increased computational overhead and system complexity.

Among the evaluated methods, the Vision Transformer (ViT) model achieved superior performance across all evaluation metrics. With an AUC-ROC score of 0.991, an F1-score of 0.966 and an overall accuracy of 97%, ViT demonstrated exceptional robustness and generalization capability. Unlike CNNs that focus predominantly on local spatial features, ViT leverages self-attention mechanisms to capture long-range dependencies and global contextual information within barcode images. This holistic understanding of image content enables ViT to maintain high recognition accuracy even under

challenging visual conditions, such as occlusion, distortion and low contrast.

Overall, the findings clearly indicate that ViT-based models are highly effective and reliable for real-world barcode identification tasks, particularly in complex and dynamic environments. The superior performance of ViT suggests its strong potential for deployment in industrial automation, logistics, inventory management and supply chain applications. Future work may focus on optimizing ViT architectures for real-time processing, reducing computational costs and extending the approach to multi-format and multi-language barcode systems to further enhance scalability and practical applicability.

Future research can focus on optimizing the Vision Transformer (ViT) model for real-time barcode identification by reducing computational complexity and inference time through model pruning and lightweight attention mechanisms. Additionally, extending the framework to support multi-format and multi-language barcodes, including damaged or partially occluded codes, would enhance its practical applicability. Integrating the system with edge and mobile devices and evaluating performance under large-scale industrial datasets can further validate robustness and scalability in real-world deployment scenarios.

Abbreviations

CNN - Convolutional Neural Networks, FN: False Negative, FP: False Positive, IDT-PI: Identifying and Detecting Text in Printable Images, OCR: Optical Character Recognition, TN: True Negative, TP: True Positive, ViT: Vision Transformers, XAI: Explainable Artificial Intelligence.

Acknowledgement

The authors have no acknowledgements to declare.

Author Contributions

Vigneshini I: conceptualization, software implementation, data curation, visualization, G Revathy: conceptualization, methodology, D Joseph Pushparaj: conceptualization, validation, visualization, SP Ramesh: conceptualization, methodology, R Saravanakumar: conceptualization, methodology, software implementation, data curation, validation, visualization, writing the original draft of the manuscript, N Raghavendran: software implementation, data curation, K Antony Sudha: data curation, validation, visualization.

Conflict of Interest

The corresponding author declares that there is no conflict of interest on behalf of all authors.

Data Availability

The dataset is available from the corresponding author on a reasonable request.

Declaration of Artificial Intelligence (AI) Assistance

No generative AI or AI-assisted technologies were used in the preparation of this manuscript.

Ethics Approval

Not Applicable.

Funding

This research did not get any funding.

References

1. Shi C, Jiang X, Zhang X, *et al*. Real-time multi-scale barcode image deblurring based on edge feature guidance. Electronics. 2025;14(7):1298. https://doi.org/10.3390/electronics14071298
2. Yang Y, Lei C, Chen R, *et al*. Fast image processing and detection of distorted, blur-readable 2D barcode. 2022 7th International Conference on Signal and Image Processing (ICSIP). 2022:322–6. https://doi.org/10.1109/icsip55141.2022.9887142
3. Von Rueden L, Mayer S, Beckh K, *et al*. Informed machine learning - a taxonomy and survey of integrating prior knowledge into learning systems. IEEE Transactions on Knowledge and Data Engineering. 2023;35(1):614-33. https://doi.org/10.1109/tkde.2021.3079836
4. Adak C. An approach for printed document labeling. IEEE First International Conference on Automation. 2014:1–4. https://doi.org/10.1109/aces.2014.6808032
5. Li D, Hou J, Gao W. Instrument reading recognition by deep learning of capsules network model for digitalization in Industrial Internet of Things. Engineering Reports. 2022;4(12):e12547. https://doi.org/10.1002/eng2.12547
6. Poon C. Factors implicating the validity and interpretation of mechanobiology studies in simulated microgravity environments. Engineering Reports. 2020;2(10):e12242. https://doi.org/10.1002/eng2.12242
7. Shawon MTR, Tanvir R, Alam MdGR. Bengali handwritten digit recognition using CNN with explainable AI. Recognition. 2022:1–6. https://doi.org/10.1109/sti56238.2022.10103341
8. Zhang Z, Damiani E, Hamadi HA, *et al*. Explainable artificial intelligence to detect image spam using convolutional neural network. 2022 International Conference on Cyber Resilience (ICCR). 2022:1-5. https://doi.org/10.1109/iccr56254.2022.9995839

9. Jain T, Sharma R, Malhotra R. Handwriting recognition for medical prescriptions using a CNN-Bi-LSTM model. 6th International Conference for Convergence in Technology. 2021:1–4. https://doi.org/10.1109/i2ct51068.2021.9418153
10. Zhang J, Li Y, Tian J, *et al*. LSTM-CNN hybrid model for text classification. 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). 2018:1675–80. https://doi.org/10.1109/iaeac.2018.8577620
11. Alheraki M, Al-Matham R, Al-Khalifa H. Handwritten Arabic character recognition for children writing using convolutional neural network and stroke identification. Human-Centric Intelligent Systems. 2023;3:147–59. https://doi.org/10.1007/s44230-023-00024-4
12. Baviskar D, Ahirrao S, Potdar V, *et al*. Efficient automated processing of the unstructured documents using artificial intelligence: A systematic literature review and future directions. IEEE Access. 2021;9:72894–936. https://doi.org/10.1109/access.2021.3072900
13. Syed R, Suriadi S, Adams M, *et al*. Robotic process automation: Contemporary themes and challenges. Computers in Industry. 2019;115:103162. https://doi.org/10.1016/j.compind.2019.103162
14. Jemni SK, Souibgui MA, Kessentini Y, *et al*. Enhance to read better: A multi-task adversarial network for handwritten document image enhancement. Pattern Recognition. 2021;123:108370. https://doi.org/10.1016/j.patcog.2021.108370
15. Abbas S, Alhwaiti Y, Fatima A, *et al*. Convolutional neural network-based intelligent handwritten document recognition. Computers, Materials & Continua/Computers, Materials & Continua. 2021;70(3):4563–81. https://doi.org/10.32604/cmc.2022.021102
16. Habibunnisha N, Sivamani K, Seetharaman R, *et al*. Reduction of noises from degraded document images using image enhancement techniques. 2019 Third International Conference on Inventive Systems and Control. 2019:522–5. https://doi.org/10.1109/icisc44355.2019.9036418
17. Chen J, Dai N, Hu X, *et al*. A lightweight barcode detection algorithm based on deep learning. Applied Sciences. 2024;14(22):10417. https://doi.org/10.3390/app142210417
18. Nanna N, Chanthawong N, Buajarern J. A two-layer deep learning-powered optical character recognition for enhance calibration processes. Measurement Sensors. 2024;38:101475. https://doi.org/10.1016/j.measen.2024.101475
19. Alzubaidi L, Bai J, Al-Sabaawi A, *et al*. A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips and applications. Journal of Big Data. 2023;10(1):1-82. https://doi.org/10.1186/s40537-023-00727-2
20. Niranjan J, AM Senthil Kumar. A novel text recognition using deep learning technique. 2024 International Conference on Advances in Data Engineering and Intelligent Computing Systems. 2024:1–6. http://dx.doi.org/10.1109/adics58448.2024.10533457
21. Wang X, Chen G, Qian G, *et al*. Large-scale multi-modal pre-trained models: A comprehensive survey. Machine Intelligence Research. 2023;20:447–82. https://doi.org/10.1007/s11633-022-1410-8
22. Li M, Lv T, Chen J, *et al*. TrOCR: Transformer-based optical character recognition with pre-trained models. Proceedings of the AAAI Conference on Artificial Intelligence. 2023;37(11):13094–102. https://doi.org/10.1609/aaai.v37i11.26538
23. Shambharkar Y, Salagrama S, Sharma K, *et al*. An automatic framework for number plate detection using OCR and deep learning approach. International Journal of Advanced Computer Science and Applications. 2023;14(4):140402. https://doi.org/10.14569/ijacsa.2023.0140402
24. Cui K, Xu Q, Ding Y, *et al*. Optical character recognition method based on YOLO positioning and intersection ratio filtering. Symmetry. 2025;17(8):1198. https://doi.org/10.3390/sym17081198
25. Khan A, Rauf Z, Sohail A, *et al*. A survey of the vision transformers and their CNN-transformer based variants. Artificial Intelligence Review. 2023;56:2917–70. https://doi.org/10.1007/s10462-023-10595-0
26. Pustokhina IV, Pustokhin DA, Rodrigues JJPC, *et al*. Automatic vehicle license plate recognition using optimal K-Means with convolutional neural network for intelligent transportation systems. IEEE Access. 2020;8:92907–17. https://doi.org/10.1109/access.2020.2993008
27. Schlüter M, Tepper C, Briese C, *et al*. Deep learning-based Optical Character Recognition for identifying on-label printed part numbers of used automotive parts: A comparative study of open source and commercial methods. In: Lecture notes in mechanical engineering; 2025:527–34. https://doi.org/10.1007/978-3-031-77429-4_58
28. Li J, Guan W. The Optical barcode detection and recognition method based on visible light communication using machine learning. Applied Sciences. 2018;8(12):2425. https://doi.org/10.3390/app8122425
29. Xiao Y, Ming Z. 1D barcode detection via integrated deep-learning and geometric approach. Applied Sciences. 2019;9(16):3268. https://doi.org/10.3390/app9163268
30. Sofia F, Sangeetha M. A comparison study on optical character recognition models in mathematical Equations and in any language. Results in Control and Optimization. 2025;18:100532. https://doi.org/10.1016/j.rico.2025.100532

How to Cite: Vigneshini I, Revathy G, Pushparaj DJ, Ramesh SP, Saravanakumar R, Raghavendran N, Sudha AK. Extraction of Optical Character Recognition for Barcode Systems using Deep Learning Techniques. Int Res J Multidiscip Scope. 2026; 7(2): 750-763. DOI: 10.47857/irjms.2026.v07i02.08859