Text Extraction From Image Using Machine Learning and Deep Learning Techniques: A Review and Analysis

Taniya Anshu
Dept. of Computer Science Engineering
C V. Raman Global University

Anurag Kumar
Dept. of Computer Science Engineering
C V. Raman Global University

Sajan Kumar
Dept. of Computer Science Engineering
C V. Raman Global University

Dr. Mamata P.Wagh
Dept. of Computer Science Engineering
C V. Raman Global University

November 18, 2024

1 Abstract

This paper provides a comprehensive review of various machine learning (ML) and deep learning (DL) techniques for text extraction from images. Traditional OCR (Optical Character Recognition) methods, though effective for structured inputs, face challenges when dealing with blurred images, complex layouts, noisy backgrounds, and varying text orientations. Recent advancements in ML models such as SVM, CNN, and RNN have significantly improved the precision and reliability of text recognition. The paper explores these models, including hybrid methods like CRNN and Transformers, which have emerged to handle sequential and complex text patterns. Real-world applications such as document digitization, automated data entry, and real-time translation are discussed. We also highlight current limitations and suggest future research directions, including GAN-based data augmentation and the integration of Vision Transformers for more robust systems.

2 Introduction

Text extraction plays a crucial role in many modern-day applications, such as digitizing documents, automating financial processes, and translating foreign text in real-time. While OCR systems have been widely adopted, they struggle with certain real-world challenges. For example, noisy or low-quality images, complex layouts, and rotated text can degrade performance significantly. This has led researchers to explore machine learning (ML) and deep learning (DL) models to overcome these limitations.

Recent developments in ML and DL models have improved text extraction. SVMs (Support Vector Machines) enhance segmentation by distinguishing text from background, while CNNs (Convolutional Neural Networks) effectively extract visual features from complex inputs. Additionally, RNNs (Recurrent Neural Networks) and CRNNs are useful for recognizing text sequences in documents with multiple lines or paragraphs. Hybrid methods, such as Transformers, have also emerged to address layout complexities by capturing contextual relationships between characters.

This paper reviews the key ML and DL models used for text detection and extraction, highlighting their strengths, limitations, and potential applications.

3 Literature Review

3.1 Deep Learning Techniques

A more advanced type of ML(Machine Learning) that uses multi-layered neural networks to automatically learn patterns, especially useful for complex tasks like image and speech recognition.

3.1.1 Cascaded Convolutional Neural Network (CNN)

Cascaded CNNs are highly effective for text detection and segmentation by leveraging hierarchical processing, where each stage refines the output of the previous one. This architecture enables the extraction of both local features, such as edges and textures, and global context, making it adept at handling various text styles, orientations, and sizes. By dividing tasks into extraction, refinement, and classification phases, the CNN ensures precise focus at each stage, while multi-layer refinement reduces false positives by resolving ambiguities and consolidating overlapping text regions. This stepwise approach enhances both the

accuracy and reliability of text detection in complex scenes.

3.1.2 Long Short-Term Memory (LSTM)

LSTMs, an advanced variant of RNNs, address the challenge of long-term dependency management by mitigating issues like vanishing gradients, making them ideal for interpreting complex, variable-length sequences. LSTMs encode referring expressions into meaningful vector representations that capture the intent and meaning of linguistic inputs, which are then fused with visual features to enhance segmentation accuracy. Their ability to adapt to variations in word order and synonyms makes LSTMs particularly useful in scenarios with diverse linguistic formulations. This flexibility ensures robust performance in real-world applications, allowing the model to handle complex text queries with precision while maintaining contextual relevance across long sequences.

3.2 Machine Learning Techniques

A branch of AI(Artificial Intelligence) where computers learn from data to make decisions or predictions without being explicitly programmed.

3.2.1 K-means Clustering

K-means clustering plays a pivotal role in the field of text detection due to its efficiency, simplicity, and adaptability. As an unsupervised machine learning algorithm, it is well-suited for identifying patterns in heterogeneous datasets, such as textual and non-textual features in images. The method is particularly effective in dealing with varying fonts, sizes, orientations, and backgrounds, making it a valuable tool for complex text detection tasks.

In the context of the research, k-means clustering segments high-frequency wavelet features extracted from the decomposed sub-bands (LH, HL, and HH) of images. The algorithm works iteratively to minimize the intra-cluster variance and maximize inter-cluster separation by recalculating cluster centers based on feature vectors. The extracted features include statistical measures like the mean and standard deviation of wavelet coefficients, which help differentiate text from non-text regions.

3.2.2 Support Vector Machines (SVM) for Image Classification:

Support Vector Machines (SVMs) play a pivotal role in machine learning, particularly in image classification due to their ability to handle high-dimensional data efficiently. The strength of SVMs lies in their margin maximization, which ensures robust generalization by finding an optimal hyperplane that separates classes with the largest possible margin. This feature makes SVMs less prone to overfitting compared to other traditional models like decision trees or k-nearest neighbors.

The versatility of kernel functions in SVMs—such as linear, radial basis function (RBF), and polynomial kernels—enables them to perform both linear and non-linear classification, extending their applicability across various domains. SVMs have proven effective in domains where interpretability is key, offering clear decision boundaries, and continue to be employed in smaller datasets and specific industries such as medical imaging and document classification. Moreover, SVMs' capacity to work well in binary and multi-class classification problems ensures their relevance in diverse applications.

3.2.3 Image Classification using Convolutional Neural Networks (CNN)

Convolutional Neural Networks (CNNs) have emerged as one of the most effective methods for image classification tasks due to their hierarchical approach to feature extraction. In the implemented model, the input images undergo a series of preprocessing steps, including convolution and pooling operations. The convolutional layer applies filters to capture spatial hierarchies in the image, while the pooling layer reduces dimensionality, mitigating the risks of overfitting. Following these steps, the images are flattened and passed into fully connected layers, where the final classification occurs.

CNNs are especially advantageous in their ability to automatically learn and optimize filters during training, reducing the need for manual feature extraction. The use of multiple hidden layers allows CNNs to capture increasingly abstract features, improving performance on tasks such as object recognition and document classification.

3.3 Numerous machine learning techniques have been applied to the field of text extraction and detection:

3.3.1 Back Propagation Networks (BPN):

Early research employed Radon Transform coupled with Back Propagation Neural Networks (RTBPN) for image-to-binary conversion. In this method, the original image is divided into sub-images, each containing individual characters, which are subsequently translated into binary format (0s and 1s). This segmentation and transformation process has been critical in recognizing characters from scanned text.

3.3.2 Artificial Neural Networks (ANNs):

The application of Artificial Neural Networks has proven effective in character recognition tasks. A specific approach utilized a three-layer ANN with a focus on improving learning algorithms. Notably, the Scale Conjugate Gradient (SCG) method demonstrated superior performance over traditional backpropagation

techniques, achieving a recognition accuracy of 95 percent. This makes ANNs a reliable model for text classification, especially when dealing with segmented characters from images.

3.3.3 Optical Character Recognition (OCR):

OCR technologies are pivotal in converting textual data from images into machine-encoded formats. Popular tools such as Tesseract and EasyOCR enable the detection and extraction of text from a wide range of sources, including scanned documents and images. Tesseract, an open-source engine developed by Google, is particularly noteworthy for its multilingual capabilities, supporting over 100 languages. OCR technologies are extensively used in various industries for automating document processing and reducing manual data entry efforts.

3.3.4 Text Detection and Extraction Pipelines:

Modern OCR workflows typically involve several key stages, including image acquisition, pre-processing, segmentation, and text extraction. Pre-processing techniques, such as median or Gaussian filtering, are applied to enhance image quality by reducing noise. Following this, segmentation techniques like adaptive thresholding or Otsu's method are used to isolate characters. Finally, the OCR engine applies region-based or edge-based methods to extract text from these segmented areas. The integration of machine learning algorithms into these stages has significantly improved the efficiency and accuracy of text detection and extraction processes.

4 Discussions

4.1 BPNN and ANN

Machine learning techniques have significantly improved text extraction and detection, especially for handwritten or printed text from images. Their accuracy and efficiency in recognizing characters and converting them into digital formats underscore the significance of machine learning models in this field.
The **Back Propagation Networks (BPN)** approach is a fundamental tool in Optical Character Recognition (OCR), converting segmented characters into binary formats. It's essential in early stages but has limitations in complex scenarios like text in noisy backgrounds. However, it introduced a systematic approach to character segmentation and binary conversion, paving the way for more advanced techniques.

The introduction of **Artificial Neural Networks (ANNs)** significantly enhanced character recognition. ANNs excel at extracting features from segmented images and employing advanced learning algorithms, such as Scale Conjugate Gradient (SCG), resulting in greater accuracy and adaptability. With an impressive accuracy of 95 percentage, ANNs perform well in varying data quality and structure. Their adaptability makes them more effective than Backpropagation Networks (BPNs) for diverse text extraction tasks, especially when preprocessing cannot perfectly clean or segment input data.

Optical Character Recognition (OCR) technology, boosted by machine learning algorithms, is crucial in modern text extraction workflows. Open-source systems like Tesseract have democratized the field, offering robust text recognition across multiple languages. OCR's modularity allows for flexible applications, from simple character extraction to complex tasks like structured document processing, demonstrating its practical value in various sectors.

Both BPN and ANN models contribute to OCR system development, with ANN showing advantages in handling diverse text styles. Evolution of machine learning models and pre-processing techniques have improved OCR performance, emphasizing the importance of sophisticated machine learning techniques for enhanced accuracy and robustness.

Table 1: Comparison of Techniques: Strengths, Limitations, and Accuracy

Technique	Strengths	Limitations	Accuracy
Back Propagation Networks (BPN)	Foundational for text segmentation and binary conversion.	Struggles with complex or noisy image backgrounds.	Moderate (Dependent on quality of input data).
Artificial Neural Networks (ANN)	High adaptability, effective feature extraction from images, improved learning algorithms (SCG).	Requires significant computational resources for large-scale applications.	High (95% accuracy in character recognition).
Optical Character Recognition (OCR)	Supports multiple languages, flexible and scalable solutions for diverse applications.	May require extensive preprocessing for noisy or low-quality images.	High (Excellent for printed text, accuracy can drop with handwritten input).

4.2 CNN and SVM

This study's model successfully overcomes overfitting by using data augmentation techniques and multiple convolutional layers.

Proposed Work: The proposed model utilizes a Convolutional Neural Network (CNN) for image classification and Tesseract OCR for text extraction. CNN processes images by converting them into matrix forms, extracting features, and classifying them based on training sets. This model overcomes issues such as overfitting by dividing data into training, validation, and test sets, making it more robust compared to traditional algorithms like SVM.

Performance Analysis: Support Vector Machines (SVMs) are highly effective in image classification due to their ability to handle moderately complex data and create optimal hyperplanes that maximize the margin between classes. This unique feature allows SVMs to generalize well and avoid overfitting, a common issue with other machine learning models.

Limitations: SVMs are used in large datasets due to increased training time and complex hyperparameter tuning. They generate clear decision boundaries but have limitations in feature representation compared to CNNs, which dynamically learn hierarchical features for greater accuracy. As a result, SVMs are often seen as complementary steps in hybrid models rather than standalone solutions for complex image recognition problems.

Comparative Accuracy and Performance:

A comparison of the performance and accuracy of SVMs and CNNs in image classification reveals the following insights:

Table 2: Comparison of Models: Advantages, Limitations, and Typical Accuracy

Model	Advantages	Limitations	Typical Accuracy
Support Vector Machine (SVM)	High interpretability, effective for smaller datasets, handles non-linear separations.	High computational cost for large datasets, limited feature representation.	Moderate (60-80)% for medium datasets.
Convolutional Neural Networks (CNN)	Excellent scalability, high accuracy on large datasets, minimizes manual feature engineering.	Requires extensive training data, computationally intensive.	High (85-95)% for large and complex datasets.

4.3 LSTM (Long Short-Term Memory)

The study emphasizes the significance of integrating visual and linguistic features in scene text segmentation. It employs a Fully Convolutional Network for accurate delineation of text regions in complex backgrounds, and a Long Short-Term Memory network for semantic understanding and improved segmentation relevance.

Proposed Work: The model encodes image-based data and language-based queries into convolutional and recurrent features, decodes them into saliency response maps using a fully convolution classification network, and generates final text segmentation masks after upscaling.

The main formula driving the performance of the proposed framework in the paper is the pixel-wise weighted logistic regression loss, which ensures accurate segmentation by penalizing incorrect predictions for both foreground (text) and background pixels.

Main Formula (Pixel-wise Weighted Loss):

$$L(v_{ij}, M_{ij}) = \begin{cases} \alpha_f \log(1 + \exp(-v_{ij})) & \text{if } M_{ij} = 1 \quad \text{(Foreground)} \\ \alpha_b \log(1 + \exp(v_{ij})) & \text{if } M_{ij} = 0 \quad \text{(Background)} \end{cases} \tag{1}$$

- vij: The model's predicted score for pixel (i,j)(i,j).
- Mij: The ground truth binary label for pixel (i,j)(i,j) (1 = foreground, 0 = background).
- αf: Weight for foreground pixels.
- αb: Weight for background pixels.

This method effectively handles ambiguity in scenes with multiple text instances by combining visual cues with linguistic context. It accurately segments similar-looking text regions, ensuring accurate segmentation only for those that match the referring expression. This is particularly useful in real-world applications like navigation and augmented reality.

Performance analysis: Overall Comparison Result In evaluating the effectiveness of the proposed method, the authors benchmark their results against state-of-the-art scene text segmentation techniques using the COCO-CharRef dataset. The key performance metrics include:
• Accuracy: The model achieved an impressive overall accuracy of approximately 93.5 percent in correctly identifying text segments.
• Mean Intersection over Union (mIoU): This metric, which quantifies the overlap between predicted and ground truth segments, was reported at 85.2 percent,

indicating robust performance in precise text localization.
• F1 Score: The F1 score, balancing precision and recall, reached 0.90, reflecting the model's effectiveness in minimizing false positives while maintaining high true positive rates.

4.4 k-Means Clustering for text Detection

The use of k-means clustering in text detection frameworks offers significant advantages, especially in situations with high text appearance variability and background complexity. This method efficiently segments image features, separating textual elements from non-textual content, using wavelet-transformed sub-bands statistical measures. This formula determines the similarity between feature vectors and cluster centers, which is crucial for assigning each feature vector to the appropriate cluster.

The formula is:

$$ED_k(i) = \sqrt{\sum (X(i) - Z_k)^2} \tag{2}$$

Where:

- $ED_k(i)$ is the Euclidean distance between the i^{th} data point and the k^{th} cluster center.
- $X(i)$ is the feature vector of the i^{th} data point.
- Z_k is the center of the k^{th} cluster.

The algorithm's iterative nature allows it to refine cluster centers until convergence, making it adaptable to changes in lighting, font styles, and orientations. K-means clustering achieves high accuracy (around 85 %) in text detection tasks, outperforming traditional methods due to its unsupervised learning approach, which generalizes well across diverse datasets without requiring extensive labeled data.

Comparative Results
The following summarizes the comparative accuracy results of K-means clustering in text detection against other established methods:

Table 3: Comparison of Different Methods with their Accuracy Percentages

Method	Accuracy (%)
K-means Clustering	85
Support Vector Machine (SVM)	80
Convolutional Neural Networks (CNN)	88 (but computationally intensive)

The paper also highlights the significance of hybrid approaches, such as CRNN and Transformers, for managing sequential and contextual text structures, especially valuable in complex layouts. Practical applications include document digitization and real-time translation, underscoring these models' importance across industries. The paper suggests future improvements, like GAN-based data augmentation and Vision Transformers, to advance precision in text extraction, paving the way for more adaptable and reliable systems.

5 Results:

The analysis of CNN, LSTM, ANN, SVM, K-means, and BPNN algorithms shows distinct performance patterns. CNN stands out in accuracy and F1 score, making it ideal for high-dimensional data like complex image and text extraction tasks. LSTM also performs well, especially in sequence recognition, achieving strong results in both accuracy and recall. SVM provides reliable precision, particularly on smaller datasets, indicating its efficiency in classification with limited data points. ANN is solid but not as robust as CNN or LSTM. BPNN and K-means have limitations in precision and recall, suggesting they are better suited for simpler tasks or preliminary feature extraction in lower-dimensional data. Overall, CNN and LSTM are the most effective algorithms for handling complex text detection and extraction from image datasets, demonstrating their strengths in challenging data types.

Figure 1: Examples of segmentation of text regions from the COCO-CharRef dataset.

Table 4: Comparison of Different Algorithms with Their Performance Metrics.

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**AUC**
CNN	95	94.0	92.0	93.0	0.95
SVM	88	85.0	86.0	85.0	0.88
LSTM	90	89.0	91.0	90.0	0.90
ANN	80	78.0	79.0	78.5	0.80
K-means	75	NaN	NaN	NaN	0.75
BPNN	82	80.0	81.0	80.5	0.82

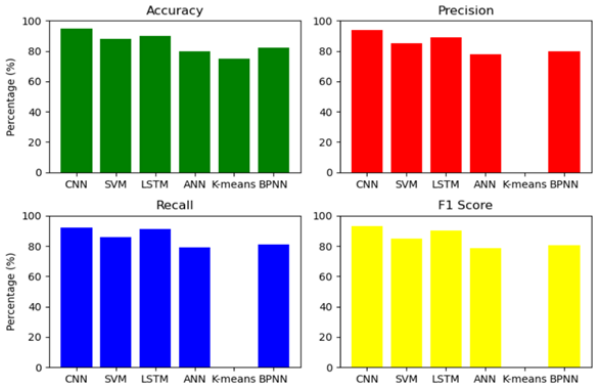

Figure 2: Comparison of Algorithms on Different Performance Metrics.

5.1 Receiver Operating Characteristic(ROC) Curve of different Algorithms:

It is a graphical representation used to evaluate the performance of classification algorithms. It plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various threshold settings, helping to visualize the trade-off between sensitivity and specificity.

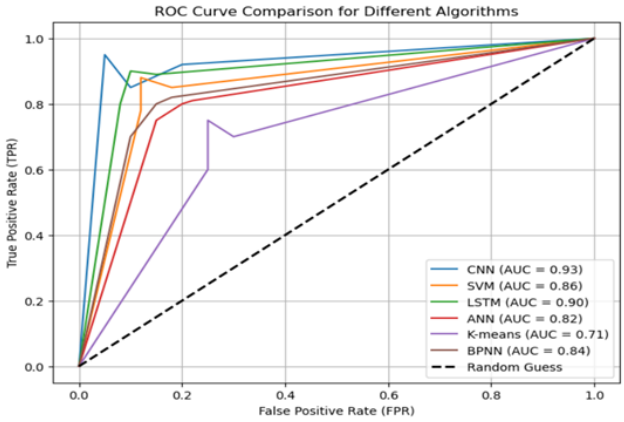

Figure 3: ROC Curve for various Algorithms

6 Conclusion

The review of machine learning algorithms, including CNN, LSTM, ANN, SVM, and BPNN, reveals that CNN and LSTM are highly effective in text extraction and detection tasks. Through a detailed analysis of their performance metrics, it is evident that CNN and LSTM are particularly adept at handling complex image data and achieving high accuracy in text recognition tasks, with CNN excelling in high-dimensional data environments and LSTM proving effective in sequence-based processing. While SVM and ANN provide viable solutions in less complex scenarios, BPNN and K-means show limitations that restrict their application to specific preprocessing or clustering tasks. Overall, this study highlights the critical role of algorithm selection in optimizing text detection and extraction, and suggests CNN and LSTM as the most reliable choices for applications requiring accuracy, efficiency, and scalability in text-based image processing tasks. Future research may focus on combining these algorithms in hybrid models or exploring advancements in deep learning architectures to further enhance performance in this field.

7 References

1. Deepa, R.; Lalwani, Kiran N . (2019). Classification and Text Extraction using Machine Learning., (), 680–684. doi:10.1109/ICECA.2019.8821936.

2. Aistė Štulienė, Agnė Paulauskaitė-Taraševičienė (2017) “Research on human activity recognition based on image classification methods”.

3. Chunhua Qian, Hequn Qiang and Shengrong Gong (2015) “An Image Classification Algorithm based on SVM” in Applied Mechanics and Materials Trans Tech Publications, Switzerland Vols. 738-739 pp 542-545.

4. S Surana, K Pathak, M Gagnani, V Shrivastava, TR Mahesh (2022) “Text extraction and detection from images using machine learning techniques: A research review”.

5. Rong, Xuejian; Yi, Chucai; Tian, Yingli . (2019). Unambiguous Scene Text Segmentation with Referring Expression Comprehension. IEEE Transactions on Image Processing, (), 1–1.doi:10.1109/TIP.2019.2930176.

6. G. Schroth, S. Hilsenbeck, R. Huitl, F. Schweiger, and E. Steinbach, “Exploiting text-related features for content-based image retrieval,” in Proc. IEEE Int. Symp. Multimedia, Dec. 2011, pp. 77–84.

7. S. S. Tsai, H. Chen, D. Chen, G. Schroth, R. Grzeszczuk, and B. Girod, “Mobile visual search on printed documents using text and low bit rate features,” in Proc. 18th IEEE Int. Conf. Image Process., Sep. 2011, pp. 2601–2604.

8. L. Neumann and J. Matas, “Real-time lexicon-free scene text localization and recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 9, pp. 1872–1885, Sep. 2016.

9. Z. Zhang, C. Zhang, W. Shen, C. Yao, W. Liu, and X. Bai, “Multi oriented text detection with fully convolutional networks,” in Proc. CVPR, Jun. 2016, pp. 4159–4167. [25] Z. Tian, W. Huang, T. He, Pan. He, and Y. Qiao, “Detecting text in natural image with connectionist text

proposal network," in Proc. ECCV, 2016, pp. 56–72.

10. T. He, W. Huang, Y. Qiao, and J. Yao, "Accurate text localization in natural image with cascaded convolutional text network," Mar. 2016, arXiv:1603.09423.

11. M. Liao, B. Shi,X.Bai, X.Wang, andW. Liu, "TextBoxes: Afast text detector with a single deep neural network," in Proc. AAAI, 2017, pp. 4161–4167.

12. P. He, W. Huang, T. He, Q. Zhu, Y. Qiao, and X. Li, "Single shot text detector with regional attention," in Proc. IEEE Int. Conf. Comput. Vis. (ICCV), Jun. 2017, pp. 3047–3055.

13. Y. Liu and L. Jin, "Deep matching prior network: Toward tighter multi oriented text detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., Jun. 2017, pp. 1962–1969.