INTERNATIONAL SCIENTIFIC JOURNAL OF ENGINEERING AND MANAGEMENT (ISJEM) ISSN: 2583-6129
CONFERENCE PROCEEDINGS: VOLUME: 05 ACME – 2026 DOI: 10.55041/ISJEM.ACME091
AN INTERNATIONAL SCHOLARLY || MULTIDISCIPLINARY || OPEN ACCESS || INDEXING IN ALL MAJOR DATABASE & METADATA

Document Image Layout Retention Techniques

Mrs.B Rupa Devi Singh,M.Tech(Ph.D),
Associate Professor
Department of AI&DS
Annamacharya Institute Of Technology and Sciences
Tirupati-517520,A.P,

rupadevi.aitt@annamacharyagroup.org

ORCID: 0009-0005-1298-737X

D Srilekha
UG Scholar,
Department of AI&DS

Annamacharya Institute Of Technology and Sciences
Tirupati-517520,A.P,
srireddy7789@gmail.com

K Ujwala Sai Sree
UG Scholar,
Department of AI&DS

Annamacharya Institute Of Technology and Sciences
Tirupati-517520,A.P,

Ujwalakamjula789@gmail.com

N Sreelatha
UG Scholar,
Department of AI&DS

Annamacharya Institute Of Technology and Sciences
Tirupati-517520,A.P,
naresreelatha0109@gmail.com

S Venkata Nagarjuna Reddy
UG Scholar,
Department of AI&DS

Annamacharya Institute Of Technology and Sciences
Tirupati-517520,A.P,

snagarjunareddy905@gmail.com

Abstract— Document layout classification is a significant task in intelligent document analysis and digital information processing. It helps computers recognize and understand the structural components of documents such as research papers, reports, and articles. The structural components of documents include text blocks, title, tables, figures, and lists. Detection of these components is critical for applications such as Optical Character Recognition (OCR), automatic document digitization, and extraction of structured data. In this research, a deep learning approach is presented using YOLOv8, a state-of-the-art and efficient object detection model, for document layout classification. Unlike conventional approaches that are rule-based or multi-stage detection, YOLOv8 detects objects in a single pass, enhancing detection speed and efficiency.

The PubLayNet dataset was employed as the main training dataset. However, since the dataset was originally in COCO format, the annotations were transformed to YOLO format to make them compatible with the YOLOv8 model. The training process was performed using resized input images with optimized hyperparameters on Apple Silicon hardware with GPU support. The performance assessment was done using conventional object detection metrics such as Precision, Recall, F1-score, and mean Average Precision (mAP).

The experimental outcome shows that the proposed method has been able to achieve a mean Average Precision (mAP@0.5) of 0.916, which is a very good detection capability at the Intersection over Union threshold of 0.5. The model performed very well in the dominant classes like text and title. But in the case of less frequent classes like figures and list, the performance was relatively lower due to imbalance in the dataset. Nevertheless, the result shows that YOLOv8 is a good trade-off between accuracy and speed.

Keywords—YOLOv8, Document Layout Analysis, Object Detection, PubLayNet

I. INTRODUCTION

The area of Artificial Intelligence (AI) is presently undergoing intense growth. The current AI models, such as ChatGPT and other language models, rely on large amounts of organized data sets for their operation. The most critical source of data is embedded in documents. Computers must be able to recognize the structure of documents before they can process the content of documents. Document layout analysis is the process of identifying all the components in a document image. The components that must be identified in this step include text paragraphs and headings and tables and figures and lists. The operation of document digitization requires this step

because it is the starting point for Optical Character Recognition (OCR). Text recognition algorithms become less accurate if they are not aware of the layout of documents. The current research work employs transformer-based models such as LayoutLM for the analysis of document layouts. The models produce successful output but require large amounts of data and high computational power. YOLO models provide better results because they process faster and require less computational power. Ultralytics created YOLOv8, which is a real-time object detection model.

II. LITERATURE REVIEW

Document layout analysis has been an extensively researched area in computer vision because of its significance in document understanding and digitization. Rule-based systems and connected component analysis were used in early document layout analysis systems. For instance, traditional approaches such as projection profiles, recursive X-Y cuts, and whitespace analysis were employed to break down the document into text and graphics regions. However, these approaches were not very effective when it came to complex page layouts or different types of documents.

The advent of machine learning made it possible for supervised learning models to learn patterns from labeled data. Region-based convolutional neural networks (R-CNN) showed that deep features were capable of enhancing the accuracy of detection beyond the capabilities of hand-engineered features. Girshick proposed Fast R-CNN, which enhanced the speed and accuracy of object detection tasks. However, these approaches were still not very efficient for real-time processing.

However, modern object detection models like YOLO (You Only Look Once) changed the paradigm by considering the detection task as a single unified regression task. YOLO models learn to predict bounding boxes and class probabilities directly from the full images. This end-to-end approach allows for faster processing and makes YOLO models suitable for real-time applications. Redmon et al. demonstrated that YOLO achieves a good balance between speed and accuracy for general object detection tasks.

More recently, the research community has concentrated on the layout analysis of document images in particular. PubLayNet was introduced as one of the largest annotated datasets for document layout detection. It consists of hundreds of thousands of scientific pages annotated with structural categories such as text, titles, tables, lists, and figures. The existence of such a large dataset has made it possible to train deep models for automatic layout recognition.

III. EXISTING SYSTEM

Document layout classification has been traditionally done through rule-based and machine learning-based systems. In the early days, document layout classification was done through heuristic methods such as projection profile analysis, connected component analysis, whitespace analysis, and X-Y cut algorithms. These methods analyzed documents based on geometric characteristics such as spacing and positioning. Although these methods were successful for simple document layouts, they performed poorly for complex or multi-column documents, and were also prone to noise and formatting changes.

Later, with the advent of deep learning, convolutional neural network (CNN)-based object detection models were developed for document layout analysis. Models such as Faster R-CNN and Mask R-CNN were developed for improved detection accuracy through automatic learning of visual features. These region-based detectors offered improved performance over traditional methods but were computationally expensive and had high training times.

Recently, transformer-based models such as LayoutLM and DocFormer have been developed for document understanding tasks. These models integrate layout knowledge with text data for improved semantic understanding. Although these models offer high accuracy, they demand large amounts of annotated data and high-performance hardware for training and testing. They are also complex to deploy in real-time settings.

YOLO (You Only Look Once) algorithms, which were earlier versions of the YOLO model, have also been used for document layout detection. The YOLO-based approach has faster inference speed and a simpler architecture compared to the region-based detectors. However, the earlier versions of the YOLO model sometimes had less accuracy in detecting small or rare layout components.

IV. PROPOSED SYSTEM

The solution involves the use of YOLOv8 for document layout classification. The solution does not involve the use of rule-based segmentation techniques or transformer

models. The solution involves the use of a single-stage detection model that detects and classifies different areas of a document without requiring the use of additional models. The YOLOv8 model is capable of detecting and classifying different areas of a document in a single pass, which enhances efficiency and speed.

The model involves the use of resized input images and predicts the bounding boxes and class probabilities. The model is capable of learning the patterns of document layouts by using pre-trained weights and fine-tuning on the converted dataset. The output of the model is the labeled bounding boxes that distinctly identify the different regions of the document layout.

Moreover, the method is hardware-efficient and can be easily deployed on devices with average computational power. Its real-time processing capability makes it suitable for automated document digitization systems, intelligent archiving systems, and data extraction systems.

In conclusion, the proposed solution offers a well-balanced framework that integrates detection accuracy, speed, and flexibility for document layout analysis.

V. METHODOLOGY

A. Dataset Preparation

The PubLayNet dataset served as the training and validation material for our research work. The researchers had to download only a specific portion of the training dataset because the dataset size exceeded their requirements. The dataset existed in COCO JSON format at its initial state. The researchers converted COCO annotations into YOLO text format because YOLOv8 needs a different annotation format. The YOLO format stores image data through two system components. The format stores each object as a separate entity through its multiple lines. The format includes class ID and normalized bounding box coordinates. The dataset conversion process created two primary folders that contained The dataset conversion process created two main folders which contained The dataset The study used in this research contained 400 annotated images.

The class distribution showed an uneven pattern because A heatmap visualization revealed that A heatmap visualization showed that A heatmap visualization showed that A heatmap visualization presented three different results A heatmap visualization showed that A heatmap visualization presented three different results A heatmap visualization presented three different results A heatmap visualization showed that A heatmap visualization presented three different results A heatmap visualization displayed three different results.The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents.

The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents The heatmap showed that document titles usually appear at the top of their respective documents.

The heatmap revealed that the title of documents is normally located at the top of the documents The heatmap revealed that the title of documents is normally located at the top of the documents The heatmap revealed that the title of documents is normally located at the top of the documents.

Fig. 1. Sample annotated images from the PubLayNet dataset showing labeled regions such as text, title, table, and figure.

B. Training Process

The model was created using the Ultralytics YOLOv8 framework, which runs in Python and uses the YOLOv8s (small) model, which strikes a balance between detection accuracy and speed. The system was able to process all the incoming images by resizing them to 640 × 640 pixels, which is the most optimal resolution for optimal system performance. The training process continued for 50 epochs, during which the model was able to learn the entire dataset from start to finish to develop its ability to detect features while ensuring the correct balance between underfitting and overfitting. The batch size of 20 was selected, which is in line with the operational capacity of MacBook Air M3, which comes with 16GB RAM. The training process utilized Apple Metal Performance Shaders, which enables the training process to be accelerated through GPU computation on Apple Silicon hardware. The training process enabled the creation of two weight files, namely best.pt, which stores the weights that generated the best validation performance, and last.pt, which stores the final weights of the model after the completion of the training process. The best.pt file was used as the benchmark for evaluation and testing since it generated the most accurate detection performance.

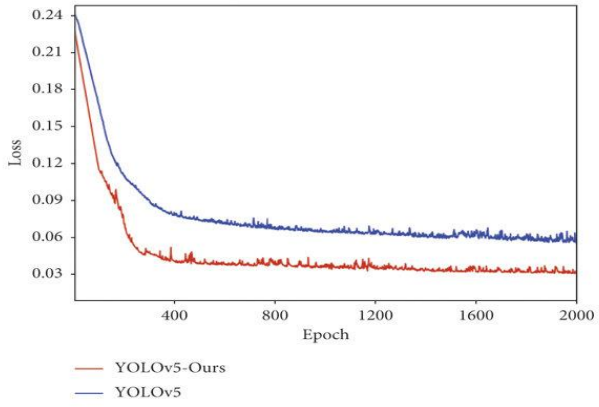

Fig. 2. Training loss curve showing reduction in loss values across epochs during model training..

VI. RESULTS AND DISCUSSION

A. Result Analysis:

The assessment of the trained model was conducted using the standard object detection metrics, which were Precision, Recall, F1-score, and Mean Average Precision (mAP). Precision is the measure of the number of correctly identified objects among the total number of predicted objects. Recall is the measure of the total actual objects that were correctly detected by the model. The F1-score is a balanced assessment that combines both Precision and Recall into a single measure. The mAP metric provides a comprehensive assessment of the model's performance. The model has an mAP@0.5 of 0.916, which is an indicator of high detection accuracy at an Intersection over Union (IoU) threshold of 0.5. The high mAP@0.5 value shows that the predicted bounding boxes have a high overlap with the ground truth annotations. The precision of different classes is as follows: Text (0.88), Title (0.93), Table (0.79), and Figure (0.86). The List class has relatively poor performance due to the small number of training data. The model has high precision for Text and Title classes because it is able to detect the main layout structures.

The detection accuracy for Table and List The confusion matrix analysis further supports these findings. The model correctly identified Text and Title classes in most cases while showing only small confusion between these two classes. The system achieves strong feature learning capabilities for categories that appear frequently in the data. The visual similarities between List elements and Text caused misclassification because there was not enough training data to differentiate between them.

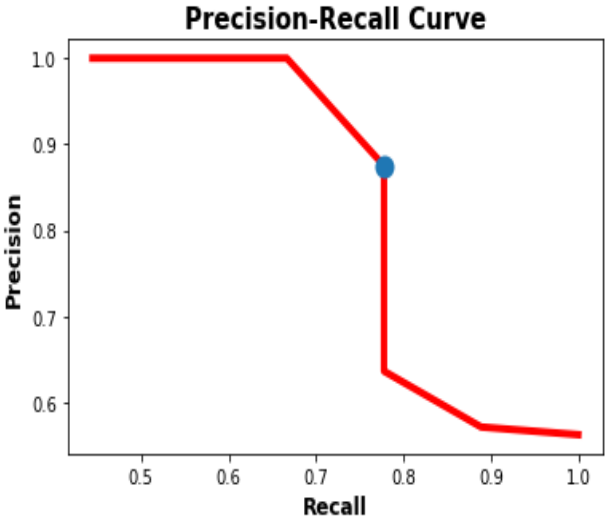

Fig. 3.Precision–Recall curves used to compute Average Precision (AP) for each class.

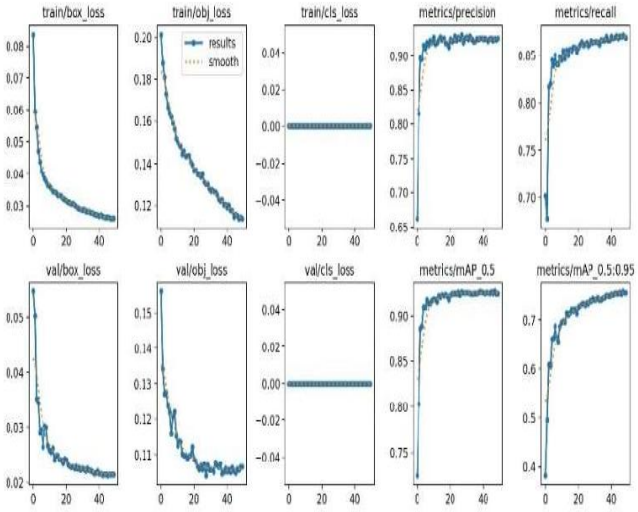

Fig. 4. Visualization of mean Average Precision (mAP@0.5) across document layout classes.

Fig. 5. Validation output illustrating classification performance of YOLOv8 for document layout elements.

VII. CONCLUSION

The researchers developed a document layout classification system through their work with YOLOv8. The researchers used YOLOv8 to test its ability to identify and categorize document elements which included text, titles, tables, figures, and lists. The researchers used the PubLayNet dataset for training after converting the annotations into OLO format. The experimental results showed that YOLOv8 achieved strong detection performance, with a mean Average Precision (mAP@0.5) of 0.916. The model showed exceptional ability to identify both text elements and title sections. The system showed decreased performance when identifying figures and lists because of the uneven distribution of classes in the dataset. The study confirms that YOLOv8 is a fast and efficient tool for document layout analysis. It needs fewer computing resources than transformer-based models while still providing high accuracy. This makes it suitable for real-time document digitization systems. The future work will increase dataset size to enhance detection capabilities for less common classes. Data augmentation techniques and more document types, like magazines, reports, and handwritten documents, can also be added to improve generalization. The integration of layout detection with OCR and language models will result in a fully automated intelligent document processing solution. The research demonstrates that YOLOv8 functions as an efficient and effective solution for automated document layout classification.

The combination of layout detection, OCR, and language models will provide a completely automated intelligent document processing system. It is clear from the research that YOLOv8 is an efficient and effective solution for the automated classification of document layouts.

VIII. FUTURE SCOPE

Although the proposed YOLOv8-based document layout classification model performed well, there are a few areas of improvement that can be pursued in future work.

First, the size of the dataset can be augmented by utilizing the entire PubLayNet dataset rather than a small sample of it. This will help in better generalization and fewer misclassifications in the underrepresented classes, such as figures and lists. Data augmentation methods such as rotation, scaling, cropping, and brightness changes can also be employed.

Second, class imbalance can be handled by employing methods such as weighted loss functions, oversampling, or synthetic data generation. These methods can help improve the detection accuracy of the minority classes.

Third, other variants of the YOLO model, such as YOLOv8m or YOLOv8l, can be experimented with to see if larger models are more accurate while still being computationally efficient.

Fourth, the proposed model can be extended to handle other types of documents, such as newspapers, magazines, invoices, handwritten documents, and archives. This will increase the applicability of the proposed model.

Fifth, the layout detection system can be combined with Optical Character Recognition (OCR) to form an end-to-end document understanding system. The layout detection system can also be combined with language models to improve semantic understanding.
Finally, future research can investigate the lightweight deployment of real-time document processing systems in mobile or cloud environments.

In conclusion, these enhancements can improve the accuracy, scalability, and usability of the proposed document layout classification system.

Apart from the above enhancements, future work may include research on enhancing the localization precision of bounding boxes through anchor box parameter fine-tuning and IoU threshold optimization. Enhanced localization can minimize overlapping detections and enhance structural simplicity in complex documents.

Another area of future research may include the application of multi-scale feature extraction methods to improve the detection of compactly arranged layout components. Certain components of the document layout, like captions and footnotes, are compact and may not be accurately detected with default parameters.

Future research may also include the application of transfer learning using domain-specific document datasets, like legal documents or scientific publications. This can enable the model to learn domain-specific layouts that are different from general research documents.

Semi-supervised or self-supervised learning methods may also be explored to minimize the need for manually annotated datasets. Manual annotation of document layouts is a time-consuming process, and minimizing this process can make the system more useful.

Cross-dataset evaluation may also be conducted to analyze the generalization capability of the model on unseen document collections. Evaluation on multiple datasets can provide stronger evidence for robustness.

REFERENCES

[1] [1] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, real-time object detection,” in Proc. IEEE CVPR, 2016, pp. 779–788.

[2] [2] J. Redmon and A. Farhadi, “YOLO9000: Better, faster, stronger,” in Proc. IEEE CVPR, 2017, pp. 7263–7271.

[3] [3] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” arXiv preprint arXiv:1804.02767, 2018.

[4] [4] W. Liu et al., “SSD: Single Shot MultiBox Detector,” in Proc. ECCV, 2016, pp. 21–37.

[5] [5] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. PAMI, vol. 39, no. 6, pp. 1137–1149, 2017.

[6] [6] T. Y. Lin et al., “Focal loss for dense object detection,” in Proc. IEEE ICCV, 2017, pp. 2980–2988.

[7] [7] A. Bochkovskiy, C. Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” arXiv preprint arXiv:2004.10934, 2020.

[8] [8] G. Jocher, A. Chaurasia, and J. Qiu, “Ultralytics YOLOv8,” Ultralytics, 2023. [Online]. Available: https://github.com/ultralytics/ultralytics

[9] [9] X. Zhong, J. Tang, and A. J. Yepes, "PubLayNet: Largest dataset ever for document layout analysis," in Proc. Int. Conf. Document Analysis and Recognition (ICDAR), 2019, pp. 1015–1022.

[10] [10] Y. Xu, M. Li, L. Cui, S. Huang, F. Wei, and M. Zhou, "LayoutLM: Pre-training of text and layout for document image understanding," in Proc. ACM SIGKDD, 2020, pp. 1192–1202.

[11] [11] J. Li et al., "DocFormer: End-to-end transformer for document understanding," in Proc. NeurIPS, 2021, pp. 3230–3242.

[12] [12] A. Das et al., "Document layout analysis: A survey," Pattern Recognit., vol.

[13] [13] G. M. Binmakhashen and S. A. Mahmoud, "Document layout analysis: A comprehensive survey," ACM Computing Surveys, vol. 52, no. 6, pp. 1–36, 2019.

[14] [14] B. Pfitzmann et al., "DocLayNet: A large human-annotated dataset for document layout segmentation," in Proc. ACM SIGKDD, 2022, pp. 3743–3751.

[15] [15] E. S. Santos Júnior, T. Paixão, and A. B. Alvarez, "Comparative performance of YOLO models for layout analysis of historical document images," Applied Sciences, vol. 15, no. 6, 2025.

[16] [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proc. IEEE CVPR, 2016, pp. 770–778.

[17] [17] L. Zhou et al., "MMDoc: A large-scale multimodal benchmark for document understanding," arXiv preprint arXiv:2201.02699, 2022.

[18] [18] X. Li, W. Yao, C. Zhang, and J. Zhang, "A hybrid CNN and transformer model for document layout analysis," IEEE Access, vol. 10, pp. 47392–47403, 2022.

[19] [19] H. Ma, G. Yin, and Q. Wu, "A deep learning method for text region extraction in document images," Pattern Recognit., vol. 114, 2021.

[20] [20] S. Gupta, R. Mehta, and M. Arora, "Deep learning based table detection and structure recognition in document images: A review," ETRI Journal, vol. 44, no. 5,