Adapting Vision Transformer-Based Object Detection Model for Handwritten Text Line Segmentation Task

Osman Furkan KARAKUŞ[1]*, Ayla GÜLCÜ[2], Ali Can KARACA[1]

[1] *Department of Computer Engineering, Yıldız Technical University, İstanbul, Turkey*

[2]*Department of Software Engineering, Bahçeşehir University, Istanbul, Turkey*

ARTICLE INFO

Received Date: 21/04/2024
Accepted Date: 27/01/2025

Cite this paper as:
Karakus, O.F., Gulcu, A. and Karaca, A.C. (2025). *Adapting Vision Transformer-Based Object Detection Model for Handwritten Text Line Segmentation Task.*Journal of Innovative Science and Engineering. 9(1): 28-38

*Corresponding author: Osman Furkan Karakuş
E-mail:osman.karakus@yildiz.edu.tr

Keywords:
Vision transformers
Handwritten text line segmentation
Object Detection
Optical Character Recognition

ABSTRACT

This study introduces a novel approach for segmenting lines of text in handwritten documents using a vision transformer model. Specifically, we adapt DEtection TRansformer (DETR) model to detect line segments in images of handwritten documents. In order to adapt DETR for the line segmentation task, we applied a pre-processing step that involves dividing each line into fixed-size image patches followed by adding positional encoding. We benefit from DETR model with a ResNet-101 backbone pretrained on the Common Objects in Context (COCO) object detection training dataset, and re-train this model using our novel, complex line segmentation dataset consisting of 1,610 handwritten forms. To evaluate the performance, another line segmentation method named Bangla Document Recognition through Instance-level Segmentation of Handwritten Text Images (BN-DRISHTI) is implemented. This method utilizes the You Only Look Once (YOLO) object detection model. Both object detection-based methods involve a learning phase during which the model is trained or fine-tuned on the dataset. For a diverse set of baselines methods, we have also implemented two learning-free algorithms such as A* Search Algorithm and the Genetic Algorithm (GA). Experimental results based on the Intersection over Union (IoU) metric demonstrate that the proposed method outperforms all other methods in terms of the detection rate, recognition accuracy, and Text Line Detection Metric (TLDM). The quantitative results also indicate that two learning-free algorithms fail to segment highly skewed lines successfully in the dataset. The A* algorithm achieves a high recognition accuracy of 0.734, compared to GA and BN-DRISHTI, which achieve recognition accuracies of 0.498 and 0.689, respectively. Our proposed approach achieves the highest recognition accuracy of 0.872, outperforming all other methods. We show that the DETR model which requires only a single fine-tuning phase for adapting to line-segmentation task, not only simplifies the training and implementation process but also improves accuracy and efficiency in detecting and segmenting handwritten text lines. DETR's use of a transformer's global attention mechanism allows it to better understand the entire context of an image rather than relying solely on local features. This is particularly beneficial for managing the diverse and complex patterns found in handwritten text where traditional models might struggle with issues such as overlapping text lines or varied handwriting styles.

1. Introduction

Handwritten document analysis has gained significant attention in the field of document image analysis and recognition, primarily due to the surge in the digitization of historical documents and the need for automated tools capable of understanding and processing handwritten texts. This process involves several steps such as pre-processing, line and word segmentation, feature extraction, and interpretation. The text line segmentation process is essential for accurate recognition in handwritten documents, as emphasized by Barakat et al. [1].

The primary challenges in segmenting lines of handwritten text stem from the diverse nature of handwriting. Variations in handwriting styles, line spacing, the presence of artifacts, and noise in scanned documents pose significant hurdles for segmentation algorithms. These complexities were effectively addressed using deep learning techniques, such as Mask R-CNN, which has demonstrated robust performance on historical documents containing various artifacts [2]. Additionally, cursive writing and overlapping text further complicate the segmentation process, requiring sophisticated methods capable of accurately identifying and separating text lines under these conditions. Many researchers have tackled this problem, and numerous methods have been introduced for efficiently segmenting and extracting lines from text documents.

The earliest applications of text segmentation methods involve learning-free statistical approaches, which have been thoroughly reviewed and detailed by Likforman-Sulem et al. [3]. Recent advancements in deep learning have enabled the use of learning-based methods for many document analysis tasks, including text line segmentation. These methods have shown significant improvements in handling the variability of handwriting styles and the complexity of text documents. Multi-dimensional Long Short-Term Memory (LSTM) Networks and Fully Convolutional Networks (FCN)-based models have been successfully utilized for line segmentation problems [4], [5], [6].

Object detection frameworks, originally designed for identifying and locating objects within images, have also been adapted to the task of text line segmentation. The application of object detection frameworks, such as Ren et al.'s Faster R-CNN [7] and Redmon et al.'s YOLO [8], to text line segmentation represents a significant shift from traditional segmentation methods. By treating text lines as objects, these frameworks leverage deep neural networks to learn from the complexities and variations present in handwritten documents and achieve remarkable accuracy in segmentation tasks. Despite the efficiency of object detection frameworks, such as Faster R-CNN and YOLO, in segmenting text lines, their generic design for general object detection poses challenges in accurately handling the specific intricacies of handwritten text, such as overlapping lines and script variability. This has led us to explore a new detection approach.

Our method aligns with the approach of BN-DRISHTI by treating text lines as objects and utilizing established object detection frameworks. However, our approach differs significantly from that study and other works in the literature. The contributions of this study, highlighting these differences, can be summarized as follows:

In this study, the DETR vision transformer model is applied to the line segmentation task for the first time. We utilized a pre-trained DETR model and adapted it specifically for the line segmentation. Fine-tuning requires image preprocessing, where each line in the dataset is divided into fixed-size image patches, followed by the addition of positional encoding. It is shown that the DETR model requires only a single-stage fine-tuning process to adapt to the line segmentation task, with no additional post-processing steps, unlike other methods such as BN-DRISHTI. In this regard, our study demonstrates that DETR not only simplifies the training and implementation process but also improves accuracy and efficiency in detecting and segmenting handwritten text lines.

DETR uses the Hungarian algorithm to optimally match the set of predicted objects with the set of ground truth objects. Once the assignment is complete, the matching loss, which combines the class prediction loss and the bounding box localization loss, is calculated. During fine-tuning, we slightly modified this combined loss to consider only two object classes: one representing a line and the other representing a non-line object.

We have selected a diverse set of methods for comparison instead of focusing on a single method. First, we used another object-detection based method, BN-DRISHTI, along with two learning-free algorithms recognized for their success in line segmentation tasks. For this purpose, we selected the A* Search Algorithm and the Genetic Algorithm.

All experiments are performed on a novel line segmentation dataset of 1,610 forms, which contains highly skewed and challenging samples compared to publicly available handwritten text datasets.

The structure of this paper is as follows: Section 2 reviews related work on handwritten text line segmentation, focusing on object detection models in this field. Section 3 details our approach, in which we adapt the DETR framework specifically for handwritten line segmentation. This section includes an overview of our Turkish Line Segmentation Dataset and the customization of vision transformers for this task. Additionally, we introduce the baseline methods for comparison, including a YOLO-based approach for Bangla line segmentation, an A path-planning method, and a Genetic Algorithm-based technique. Section 4 presents the results and discussion. Finally, Section 5 concludes the paper and outlines future research directions.

2. Related Work

The task of handwritten text line segmentation has been extensively studied, with approaches ranging from tradi-tional image processing techniques to advanced machine learning algorithms. Projection profile-based approaches have been the most widely used methods due to their simplicity; however, horizontal projections cannot handle skewed, curved, or fluctuating lines. Many improvements to these methods have been proposed in the literature, such as [9], [10], and [11]. [12] provide an overview of text line segmentation methods.

There are also heuristic-based approaches that analyze the structural properties of handwritten texts, such as the spaces between lines and the alignment of text to segment lines [13]. While effective for documents with clear and consistent handwriting, these methods often struggle with cursive or overlapping text and documents containing noise and artifacts. Surinta et al. introduced an innovative approach to line segmentation of handwritten documents using the A path-planning algorithm, which employs soft cost functions for separating text fields. This method addresses the challenge of overlapping text by calculating near-optimal paths and demonstrates effective application on historical and contemporary manuscripts with minimal adjustments required for implementation [14].

Toiganbayeva et al. introduced the Kazakh Offline Handwritten Text Dataset (KOHTD), significantly enriching Handwritten Text Recognition (HTR) research, especially for the Kazakh language. This dataset, notable for its extensive collection, underpins various HTR methodologies, showcasing adaptability through both traditional and contemporary models. A highlight of their work is the innovative application of a GA for line and word segmenta-tion, streamlining the process with improved precision and efficiency [15].

With the rise of deep learning, researchers have shifted towards data-driven approaches, applying Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to learn complex patterns in handwritten docu-ments for segmentation tasks. These methods have shown significant improvements in handling the variability of handwriting styles and the complexity of historical documents. Graves et al. [16] presented an innovative RNN model for unconstrained handwriting recognition that employs Connectionist Temporal Classification (CTC) for direct mapping from input sequences to labels without requiring pre-segmentation, demonstrating the potential of RNNs in handling complex pattern recognition tasks. Voigtlaender et al. [17] advanced handwriting recognition by employing multidimensional long short-term memory (MDLSTM) networks, showcasing their study’s ability to achieve state-of-the-art results on handwriting databases. Their work emphasizes the importance of deep network architectures and introduces an efficient GPU-based implementation, highlighting the significant impact of MDLSTM networks in the field. Moysset et al. [6] use a similar model for handwritten text line location problem of the Maurdor database [18] which is a multi-lingual database (French, English, Arabic) with both handwritten documents and printed documents. Barakat et al. [5] provide a document dataset with multi-skewed, multi-directed and curved handwritten text lines and apply line segmentation using FCNs. Renton et al. [4] propose an-other line segmentation method based on FCNs with dilated convolutions.

While CNNs and RNNs have revolutionized the field of handwritten document segmentation by learning complex patterns directly from data, these methods exhibit certain limitations. For instance, CNNs and RNNs typically re-quire extensive data preprocessing and augmentation to effectively handle the variability in handwriting styles. They often depend on large, annotated datasets and substantial computational resources for training, which can be prohibitive. Additionally, methods such as those proposed by Graves et al. [16] and Voigtlaender et al. [17], while effective, might struggle with real-time applications due to the computational demands of RNNs, particularly MDLSTM networks. Moreover, while FCNs, as used by Barakat et al. [5] and Renton et al. [4], provide robust segmentation capabilities, they may suffer from issues related to scale and translation invariance due to their fully convolutional nature.

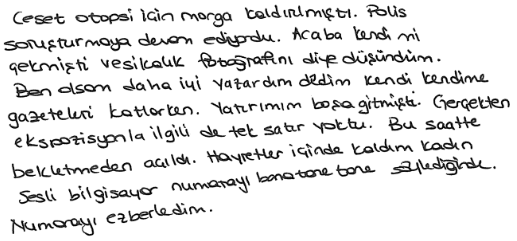
Recent studies have explored the adaptation of object detection models for text line segmentation, demonstrating promising results in accurately segmenting lines of text from a various handwritten document [19]. Baek et al. [20] introduced a novel approach to character region awareness in text detection, significantly advancing the field. Furthermore, Qu et al. [21] provided insights into robust tampered text detection in document images, presenting both a new dataset and a solution to enhance document security and integrity. Jubaer et al. [22] introduce a novel method called BN-DRISHTI for the recognition of handwritten Bangla text documents. By integrating the YOLO framework with Hough and Affine transformations for skew correction, they achieve state-of-the-art segmentation results. Their method's effectiveness is further validated on several external datasets, showcasing superior performance on unseen samples [22].

3. Materials and Methods

Our approach to handwritten text line segmentation utilizes the architecture of vision transformers, a paradigm shift in leveraging self-attention mechanisms to process images, which has shown remarkable success in various computer vision tasks. In this study, we adapt the DETR [23] model for the handwritten text line segmentation task. By adapting this architecture, we propose a novel strategy for segmenting handwritten text lines by training the model to predict ground truth text lines bounding boxes instead of traditional object bounding boxes. In this section, we first briefly mention the datasets used in this study. Then, we provide details about the adaptation of a vision transformer-based object detection method for the line segmentation task. Finally, we introduce the baseline methods employed for performance comparison.

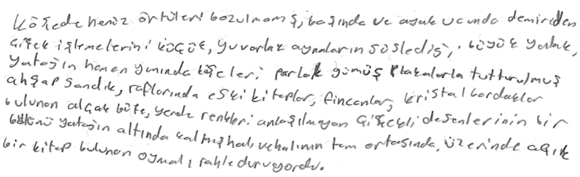
3.1. Turkish Line Segmentation Dataset

In this paper, we created a dataset for the line segmentation task by collecting handwriting samples from various authors using excerpts from 14 distinct literary books. Participants were given A4-sized forms with one of these excerpts printed at the top and were instructed to transcribe it by hand. The area where participants transcribed the excerpt was enclosed by two fixed solid lines that were later used to extract the handwriting. No lines were provided to guide participants' handwriting, resulting in highly skewed entries that are considered challenging samples, as illustrated in Figures 1 and 2. In the figures, the lines exhibit non-uniform orientations and variable spacing. In this regard, this dataset is instrumental for evaluating the performance of segmentation algorithms when faced with real-world handwriting variations and irregularities and facilitates a robust evaluation framework for comparing the effectiveness of various segmentation algorithms.



Ceset otopsi için morga kaldırılmıştı. Polis
soruşturmaya devam ediyordu. Acaba kendi mi
çekmişti vesikalık fotoğrafını diye düşündüm.
Ben olsam daha iyi yazardım dedim kendi kendime
gazeteleri katlarken. Yatırımım boşa gitmişti. Gerçekten
ekspozisyonla ilgili de tek satır yoktu. Bu saatte
bekletmeden açıldı. Hayretler içinde kaldım kadın
Sesli bilgisayar numarayı bana tane tane söylediğinde.
Numarayı ezberledim.

Figure 1: A hard example from our Dataset.



Köşede henüz örtüleri bozulmamış, başında ve ayak ucunda demirden
çiçek işlemelerini küçük, yuvarlak aynaların süslediği, büyük yatak,
yatağın hemen yanında köşeleri parlak gümüş plakalarla tutturulmuş
ahşap sandık, raflarında eski kitaplar, fincanlar, kristal bardaklar
bulunan alçak büfe, yerde renkleri anlaşılmayan çiçekli desenlerinin bir
bölümü yatağın altında kalmış halı, ve halının tam ortasında, üzerinde açık
bir kitap bulunan oymalı rahle duruyordu.

Figure 2: Another hard example from our Dataset.

The dataset comprises 1,610 forms written by 183 individuals, resulting in an average of approximately 9 forms per author. A semi-manual labeling approach was used to generate the ground truth line segments for each form. Each form was cross-checked to ensure the integrity of the dataset.

3.2. Vision Transformer for Line Segmentation

In this section we first provide a brief description on the vision transformers. Then, we present the details regarding fine-tuning process of the selected vision transformer.

3.2.1. Vision Transformers

Transformers have shown remarkable performance in natural language processing (NLP) due to their powerful self-attention mechanism. Given their significant success in NLP, researchers have started exploring how Transformers can be used in computer vision (CV). Although CNNs have long been the backbone of vision tasks, Transformers are increasingly proving to be a strong alternative. They are being applied not only to image classification but also to tasks such as object detection, semantic segmentation, and even video analysis. Unlike traditional CNNs that analyze images through localized filters, vision transformers treat an image as a sequence of patches and apply self-attention across these patches. This approach allows the model to capture global dependencies across the image, making it particularly well-suited for identifying the nuanced patterns of handwritten text lines.

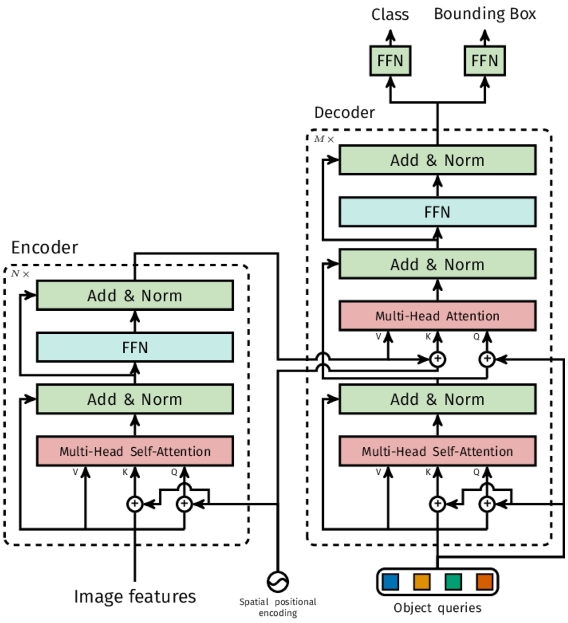


Figure 3: Architecture of DETR Model [23].

In recent years, new transformer-based vision models have emerged at a rapid pace. Han et al. [24] provide an extensive review of vision transformers. DETR is one of the most successful vision transformer models for the object detection task. DETR, an end-to-end object detector, approaches object detection as a straightforward set prediction problem, removing the need for traditional hand-crafted components like anchor generation and non-maximum suppression (NMS) post-processing. The process begins with a CNN backbone to extract features from the input image. After fixed positional encodings are added to the flattened features, these vectors are fed into the Transformer's encoder-decoder. Each encoder layer includes a multi-head self-attention module and a feed-forward network (FFN). The decoder then takes N learned positional encodings, known as object queries, as input (see Figure 3). It additionally attends to the encoder output to produce N output embeddings. Here, N is a predefined parameter, typically set larger than the number of objects expected in an image. FFNs are used to compute the final predictions, which include bounding box coordinates and class labels to specify the object class. Unlike the original Transformer, DETR decodes N objects in parallel. It employs a bipartite matching algorithm to align predicted objects with ground-truth objects and uses the Hungarian loss to calculate the loss function across all matched pairs.

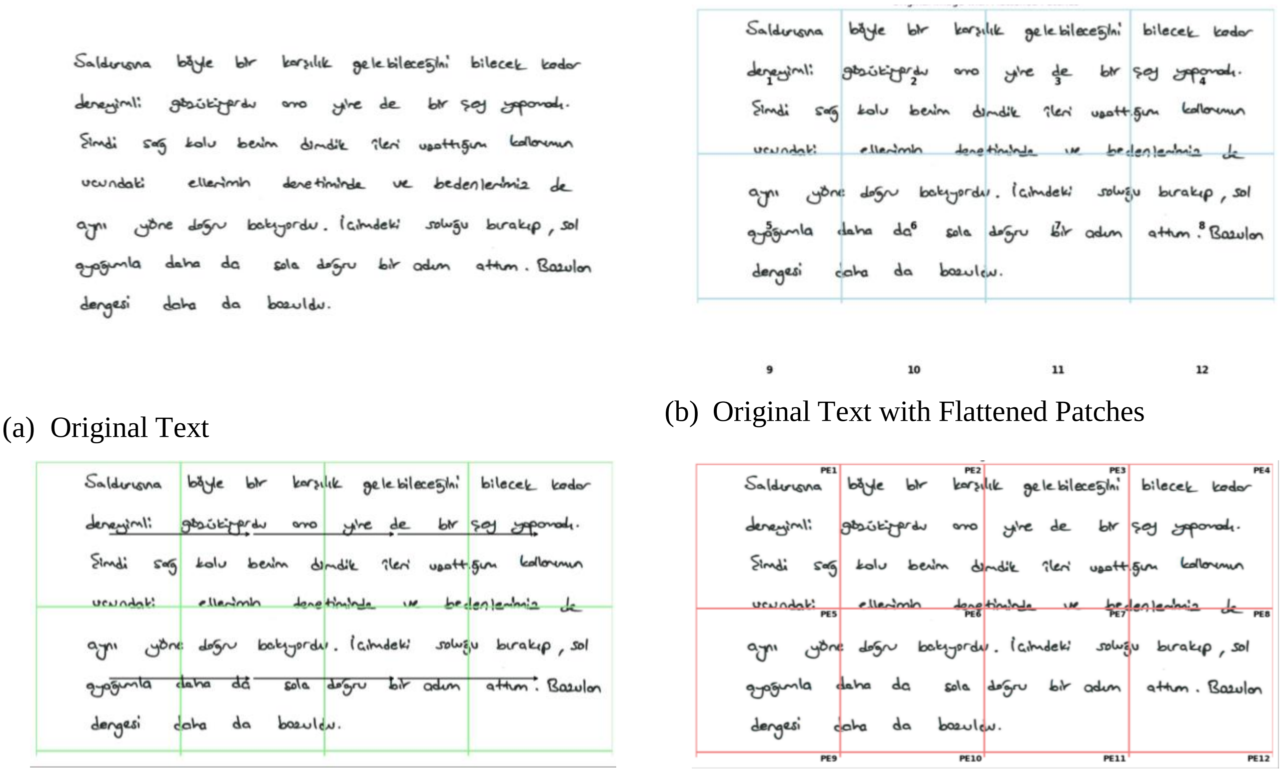
3.2.2. Adaptation for Line Segmentation Task

We use a DETR model with a ResNet-101 backbone pretrained on the COCO object detection training dataset [coco2024], which contains more than 200,000 images and 80 object categories. We fine-tune this model using our line segmentation dataset. The transformer encoder processes sequences of patch embeddings, allowing the model to learn contextual relationships between different parts of the image. The key adaptation of the proposed approach lies in the training process. Instead of training the model to identify generic objects, we train it specifically to recognize and predict the bounding boxes of text lines.

To adapt the vision transformer for line segmentation, we first pre-processed the input images by dividing them into fixed-size patches. These patches are then flattened and passed through a linear projection layer, along with positional encoding to retain information about each patch's location within the image. These image preprocessing steps are illustrated in Figure 4.

We fine-tuned the DETR model using our fully annotated dataset of 1,610 forms. A total of 100 pages are reserved for testing. We ensured that the test split included pages with varying levels of difficulty, ranging from particularly challenging images to those that were somewhat easier. A key aspect of DETR is its unique loss function, which employs bipartite matching via the Hungarian algorithm. This algorithm efficiently computes the optimal assignment that minimizes matching loss between the set of predicted objects and the set of ground truth objects. Once each ground truth object in each image is assigned a prediction, the matching loss, accounting for both class prediction and the similarity between predicted and ground truth bounding boxes, is calculated. We adopted this combined loss function during the fine-tuning process, though the number of classes was reduced to two: one representing a line and the other a non-line object.

For the training parameters, we adopt most of the values used to train DETR initially. The AdamW optimizer with a learning rate of 0.0001 was used, and a batch size of 16 was adopted. The pretrained DETR model was fine-tuned for 50 epochs using a dataset of approximately 1,500 pages on a single A100 GPU. DETR fine-tuning steps are given in the pseudo-code in Algorithm 1.



(a) Original Text

(b) Original Text with Flattened Patches

(c) Linear Projection Presented on Patches

(d) Positional Encoding Shown on Patches

Figure 4: Image preprocessing steps: original image (a), image with fixed size patches and flattening (b), linear projectionapplied on flattened image patches (c), positional embedding added to image patches (d).

Algorithm 1: DETR: End-to-End Object Detection with Transformers

```
Require: Image set $S = \{I_1, I_2, \ldots I_n\}$
Ensure: Detected objects with class labels and bounding boxes for each image
   Initialize CNN backbone and Transformer encoder-decoder architecture
   Initialize a fixed set of learned object queries
   for each image $I$ in $S$ do
Extract feature map $F$ from $I$ using the CNN backbone
Flatten $F$ into a sequence of feature vectors
         Add positional encodings to the sequence
         Pass the sequence through the Transformer encoder to obtain encoded features
         Pass encoded features and object queries through the Transformer decoder
for each output embedding from the decoder do
   Apply FFN to predict class label and bounding box
            Store the predicted class label and bounding box
end for
   end for
return Predicted class labels and bounding boxes for all images
```

3.3. Baseline Methods

To better assess the effectiveness of the proposed approach, we employ another object detection framework, YOLO, which was previously utilized in [22] for a similar line segmentation task. Since this method, like our DETR-based approach, is learning-based and requires a rigorous training phase, we selected two learning-free algorithms recognized for their success in line segmentation tasks. For this purpose, we chose the A Search Algorithm and the Genetic Algorithm, implementing each. Each of these three benchmarking methods is explained in this section.

3.3.1. YOLO-based Bangla Line Segmentation Method

Jubaer et al. [22] propose integrating of the YOLO deep learning-based object detection framework with skew correction techniques for Bangla Handwriting Segmentation. They name their method BN-DRISHTI which stands for Bangla Document Recognition through Instance-level Segmentation of Handwritten Text Images. Their approach is shown in Algorithm 2.

Algorithm 2: BN-DRISHTI

```
Require: Set of Bangla handwritten document images
Ensure: Segmented lines with bounding boxes
    Load pre-trained YOLO model for object (text line) detection
    for each image in the input set do
        Apply preprocessing (e.g., skew correction) on the image
        Detect text lines using YOLO model
        for each detected line do
            Calculate and store bounding box
        end for
    end for
return All detected lines with their bounding boxes
```

3.3.2. A* Path Planning

The A* Path Planning algorithm adapts the classic pathfinding technique to navigate through the intricacies of handwritten text segmentation. This adaptation prioritizes efficient traversal of text regions, guided by cost functions designed to distinguish between upper and lower text boundaries [14]. Please see Algorithm 3 for a step-by-step explanation.

Algorithm 3: A* Path Planning for Line Segmentation

```
Require: Start node, Goal node
Ensure: Path from Start to Goal
    Initialize OpenSet with Start node
    Initialize ClosedSet as empty
    while OpenSet is not empty do
        Current ← node in OpenSet with lowest f-score
        if Current is Goal then M
            return path reconstructed from Current
        end if
        Move Current from OpenSet to ClosedSet
        for each neighbor of Current do
            if neighbor is in ClosedSet then
                continue
            end if
            if neighbor is not in OpenSet then
                Add neighbor to OpenSet
            end if
            Update neighbor's scores based on Current
        end for
    end while
```

3.3.3. Genetic Algorithm-based Line Segmentation

The Genetic Algorithm employs evolutionary strategies to optimize the segmentation task. Through selection, crossover, and mutation, it iteratively refines solutions and converges on an optimal segmentation strategy [15]. The steps of this method are outlined in Algorithm 4.

Algorithm 4 Genetic Algorithm for Line Segmentation

```
Require: Initial population, Fitness function
Ensure: Optimal individual representing segmentation solution
    Generate initial population randomly
    while termination condition is not met do
        Evaluate fitness of each individual
        Select individuals for reproduction
        Crossover selected individuals to create offspring
        Mutate offspring with a given probability
        Select individuals for the next generation
    end while
    return the best individual from final generation
```

4. Results and Discussion

In this section, we first briefly describe the performance evaluation metrics used in the study, followed by the experimental results. Our first evaluation metric, IoU, quantifies the overlap between predicted bounding boxes and ground truth. IoU, the primary evaluation metric for object detection algorithms, is calculated as the area of overlap divided by the area of the union between the predicted and ground truth bounding boxes. An IoU threshold was established to classify predictions as accurate, facilitating a direct comparison of the methods' efficacy based on their average IoU scores across the dataset.

We also calculate the detection rate, recognition accuracy, and TLDM as detailed by Louloudis et al. [25]. The detection rate evaluates how well the proposed method can identify individual text lines in each document. For each line, a matching score is assigned based on the proportion of the predicted pixels that fall into the ground truth region. All the lines with a score above a threshold are accepted as a match. The detection rate is then computed as the proportion of correctly identified text lines out of the total number of ground-truth text lines in the document. Recognition accuracy, on the other hand, is computed as the proportion of correctly identified text lines out of the total number of detected text lines in the document. In this regard, one can consider the detection rate as recall and the recognition accuracy as precision. TLDM combines detection rate and recognition accuracy to provide a balanced measure of overall performance. It is calculated as the harmonic mean of detection rate and recognition accuracy, similar to the F1-score in traditional classification metrics.

The segmentation performance of each of the four methods, measured by average IoU, is presented in Table 1. These results indicate that our approach, which adapts vision transformer-based object detection methods for handwritten text line segmentation, outperforms all other baseline methods. Table 2 provides a comparison of all methods based on detection rate, recognition accuracy, and TLDM. The experimental results suggest that our approach achieves the highest scores across all metrics.

Table 1: Average IoU comparison of text line segmentation methods.

Method	Average IoU
A* Path Planning:	0.298
Genetic Algorithm:	0.655
BN-DRISHTI:	0.762
Our Approach:	0.925

Table 2: Accuracy-based performance comparison of the line segmentation methods

Method	Detection Rate	Recognition Accuracy	TLDM
A* Path Planning:	0.2485	0.7340	0.3715
Genetic Algorithm:	0.655	0.4978	0.4940
BN-DRISHTI:	0.762	0.6890	0.6830
Our Approach:	0.925	0.8720	0.8610

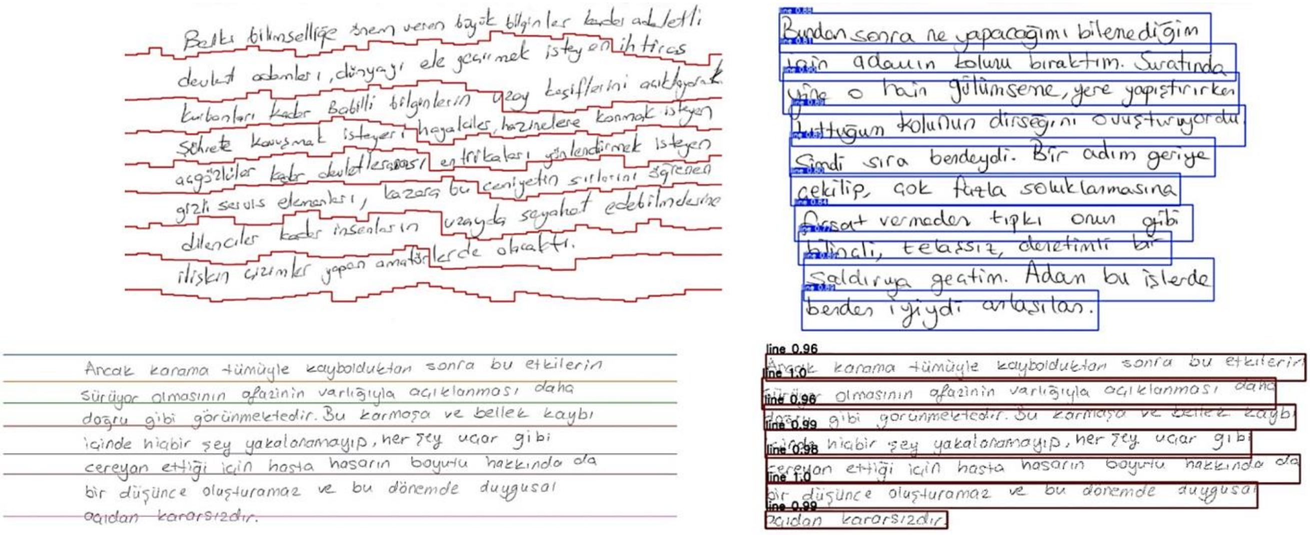

Bundan sonra ne yapacağımı bilemediğim
için adamın kolunu bıraktım. Suratında
yine o hain gülümseme, yere yapıştırırken
tuttuğum kolunun dirseğini oluşturuyordu.
Şimdi sıra bendeydi. Bir adım geriye
çekilip, çok fazla soluklanmasına
fırsat vermeden tıpkı onun gibi
dinçli, telaşsız, dirençli bir
saldırıya geçtim. Adam bu işlerde
benden iyiydi anlaşılan.
Ancak karama tümüyle kaybolduktan sonra bu etkilerin
sürüyor olmasının afazinin varlığıyla açıklanması daha
doğru gibi görünmektedir. Bu karmaşa ve bellek kaybı
içinde hiçbir şey yakalanamayıp, her şey uçar gibi
cereyan ettiği için hasta hasarın boyutu hakkında da
bir düşünce oluşturamaz ve bu dönemde duygusal
açıdan kararsızdır.
line 0.96
line 0.98
line 0.99
line 0.98
line 1.0
line 0.99

Figure 5: Visualization of the results of line segmentation algorithms: Genetic Algorithm (a), BN-DRISHTI (b), A* Path Planning (c), and our Vision Transformer-based approach (d).

In addition, we visualized the results of the line segmentation methods in Figure 5. Here, the Genetic Algorithm appears to have some difficulty accurately segmenting the lines, as indicated by several misalignments. This suggests that the algorithm may require a more comprehensive hyperparameter tuning process for better performance. The BN-DRISHTI method shows better performance than the Genetic Algorithm but still makes some errors in line segmentation, particularly at points where the line curves or where text is closely packed. The A* algorithm, which is generally efficient in many search scenarios, performed moderately well, although there are areas where the segmentation cuts through the text. The effectiveness of the A* algorithm in this context heavily relies on how the problem is framed as a graph search and how the heuristic is designed. The segmentation results in the figure suggest that our approach provides the cleanest line segmentation with no visible errors. Specifically, our approach solves the misalignment and overlapping problems through the text.

5. Conclusion

In this study, we introduce a novel approach for segmenting lines of text in handwritten documents using a vision transformer model. Specifically, we adapt the DETR model, recognized for its state-of-the-art performance in object detection, to detect line segments in images of handwritten documents. For comparison, another line segmentation method based on an object detection framework is included: the BN-DRISHTI method, which utilizes the YOLO object detection model. Both object detection-based methods involve a learning phase, during which the model is trained or fine-tuned on the dataset. Additionally, we selected two learning-free algorithms from the literature that have been successfully applied to line segmentation tasks and included them in the comparison.

Experimental results based on the Intersection over Union (IoU), one of the most widely used performance evaluation metrics for object detection, demonstrate that our method outperforms all other methods. The quantitative results further support these findings. Our line segmentation dataset primarily contains highly skewed and challenging samples, highlighting that the learning-free algorithms, A* and GA, fail to successfully segment these lines. Additionally, the BN-DRISHTI method, which is specifically designed and trained for Bangla handwriting, is less effective compared to our approach. In terms of the detection rate, recognition accuracy, and TLDM, our approach outperforms all baseline methods across each criterion. Experimental results demonstrate that, the A* algorithm achieves a high recognition accuracy of 0.734, compared to GA, BN-DRISHTI, and our approach, which achieve recognition accuracies of 0.4978, 0.689 respectively. Our proposed approach achieves the highest recognition accuracy of 0.872, outperforming all other methods. This demonstrates that A* has higher precision than the other learning-free algorithm, GA. However, in terms of the detection rate, GA outperforms A*. Given the relatively poor performance of these two learning-free algorithms on this challenging dataset, evaluating their performance on other test datasets with less complex samples would be beneficial. Investigating datasets where these methods perform well would provide valuable insights, as they require no learning phase and are therefore less costly than learning-based methods. This consideration should also be addressed in future studies. Accordingly, we plan to compare these four methods across line segmentation datasets of varying difficulty in future work.

In conclusion, this study represents a notable advancement in the field of handwritten text line segmentation through the application of a vision transformer model. DETR's use of a transformer's global attention mechanism allows it to better understand the entire context of an image, rather than relying solely on local features. This is particularly beneficial for handling the diverse and complex patterns found in handwritten text, where traditional models might struggle with issues such as overlapping text lines or varied handwriting styles. Our experimental results confirm that DETR not only simplifies the training and implementation process but also improves accuracy and efficiency in detecting and segmenting handwritten text lines. With the growing demand for digitizing handwritten texts, the development of robust segmentation techniques has become increasingly essential. This research thus provides a valuable contribution to the domain of document image analysis and recognition, promoting more efficient and comprehensive digitization processes.

In our future work, we plan to conduct cross-domain evaluations of line segmentation methods that exploit domain relations across different datasets. Furthermore, we plan to work on an end-to-end text recognition task, which is crucial for properly utilizing segmented lines and words.

Article Information Form

Funding

The authors have no received any financial support for the research, authorship or publication of this study.

Authors' Contrtibution

The authors confirm sole responsibility for the study.

The Declaration of Conflict of Interest/Common Interest

No conflict of interest or common interest has been declared by the authors.

The Declaration of Ethics Committee Approval

This study does not require ethics committee permission or any special permission.

References

[1] Barakat, K., Berat, Rafi Cohen, Ahmad Droby, Irina Rabaev, and Jihad El-Sana. "Learning-free text line segmentation for historical handwritten documents." Applied Sciences 10, no. 22 (2020): 8276.

[2] Droby, A., Barakat, B., Alaasam, R., Madi, B., Rabaev, I., & El-Sana, J. "Text Line Extraction in Historical Documents Using Mask R-CNN." Signals, vol. 3, pp. 535–549, Aug. 2022, doi: 10.3390/signals3030032.

[3] Likforman-Sulem, L., Zahour, A., & Taconet, B. "Text Line Segmentation of Historical Documents: A Survey." International Journal on Document Analysis and Recognition (IJDAR), vol. 9, May 2007, doi: 10.1007/s10032-006-0023-z.

[4] Renton, Guillaume, Yann Soullard, Clément Chatelain, Sébastien Adam, Christopher Kermorvant, and Thierry Paquet. "Fully convolutional network with dilated convolutions for handwritten text line segmentation." International Journal on Document Analysis and Recognition (IJDAR) 21 (2018): 177-186.

[5] Barakat, Berat, Ahmad Droby, Majeed Kassis, and Jihad El-Sana. "Text line segmentation for challenging handwritten document images using fully convolutional network." In 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 374-379. IEEE, 2018.

[6] Moysset, Bastien, Christopher Kermorvant, Christian Wolf, and Jérôme Louradour. "Paragraph text segmentation into lines with recurrent neural networks." In 2015 13th international conference on document analysis and recognition (ICDAR), pp. 456-460. IEEE, 2015.

[7] Ren, S., He, K., Girshick, R., & Sun, J. "Faster R-CNN: Towards real-time object detection with region proposal networks." CoRR, vol. 28, 2015.

[8] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. "You Only Look Once: Unified, Real-Time Object Detection." CoRR, vol. abs/1506.02640, 2015.

[9] Arivazhagan, Manivannan, Harish Srinivasan, and Sargur Srihari. "A statistical approach to line segmentation in handwritten documents." In Document recognition and retrieval XIV, vol. 6500, pp. 245-255. SPIE, 2007.

[10] Sanasam, Inunganbi, Prakash Choudhary, and Khumanthem Manglem Singh. "Line and word segmentation of handwritten text document by mid-point detection and gap trailing." Multimedia Tools and Applications 79, no. 41 (2020): 30135-30150.

[11] dos Santos, Rodolfo P., Gabriela S. Clemente, Tsang Ing Ren, and George DC Cavalcanti. "Text line segmentation based on morphology and histogram projection." In 2009 10th International Conference on Document Analysis and Recognition, pp. 651-655. IEEE, 2009.

[12] Louloudis, Georgios, Basilios Gatos, Ioannis Pratikakis, and Constantin Halatsis. "Text line and word segmentation of handwritten documents." Pattern recognition 42, no. 12 (2009): 3169-3183.

[13] Smith, R. "An overview of the Tesseract OCR engine." In Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), vol. 2, 2007, pp. 629–633, IEEE.

[14] Surinta, O., Holtkamp, M., Karabaa, F., Van Oosten, J.-P., Schomaker, L., & Wiering, M. "A Path Planning for Line Segmentation of Handwritten Documents." In 2014 14th International Conference on Frontiers in Handwriting Recognition, pages 175-180, IEEE, 2014.

[15] Toiganbayeva, N., Kasem, M., Abdimanap, G., Bostanbekov, K., Abdallah, A., Alimova, A., & Nurseitov, D. "KOHTD: Kazakh offline handwritten text dataset." Signal Processing: Image Communication, vol. 108, pages 116827, Elsevier BV, Oct. 2022.

[16] Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., & Schmidhuber, J. "A novel connectionist system for unconstrained handwriting recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 5, pp. 855–868, 2009.

[17] Voigtlaender, P., Doetsch, P., & Ney, H. "Handwriting Recognition with Large Multidimensional Long Short-Term Memory Recurrent Neural Networks." In 2016 15th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 228–233, 2016.

[18] Brunessaux, Sylvie, Patrick Giroux, Bruno Grilheres, Mathieu Manta, Maylis Bodin, Khalid Choukri, Olivier Galibert, and Juliette Kahn. "The maurdor project: Improving automatic processing of digital documents." In 2014 11th IAPR international workshop on document analysis systems, pp. 349-354. IEEE, 2014.

[19] Long, S., He, J., Yao, C., Hu, W., Wang, Q., & Bai, X. "TextSnake: A Flexible Representation for Detecting Text of Arbitrary Shapes." CoRR, vol. abs/1807.01544, 2018.

[20] Baek, Y., Lee, B., Han, D., Yun, S., & Lee, H. "Character Region Awareness for Text Detection." CoRR, vol. abs/1904.01941, 2019.

[21] Qu, C., Liu, C., Liu, Y., Chen, X., Peng, D., Guo, F., & Jin, L. "Towards Robust Tampered Text Detection in Document Image: New Dataset and New Solution." In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2023, pp. 5937–5946.

[22] Jubaer, S. M., Tabassum, N., Rahman, M. A., & Islam, M. K. "BN-DRISHTI: Bangla Document Recognition Through Instance-Level Segmentation of Handwritten Text Images." In Document Analysis and Recognition – ICDAR 2023 Workshops, Mickael Coustaty and Alicia Fornés, Eds., Springer Nature Switzerland, Cham, pages 195–212, 2023.

[23] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. "End-to-End Object Detection with Transformers." CoRR, vol. abs/2005.12872, 2020.

[24] Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y. and Yang, Z., 2022. A survey on vision transformer. IEEE transactions on pattern analysis and machine intelligence, 45(1), pp.87-110.

[25] Louloudis, Georgios, Basilios Gatos, Ioannis Pratikakis, and Constantin Halatsis. "Text line detection in handwritten documents." Pattern recognition 41, no. 12 (2008): 3758-3772.