Journal of Harbin Engineering University
ISSN: 1006-7043
Vol 45 No. 8
August 2024

Classification of Text and Non-Text from Bilingual Document Images Using Deep Learning Approach

Shivakumar G[1] ,Ravikumar M[2] ,Shivaprasad B J[2]
[1] Assistant professor, Dept. of IT & Computer Application Vignan's University, Guntur, AndhraPradesh-India.
[2] Professor Dept. of Computer Science, Kuvempu University, Jnanasahyadri, Shivamogga, India.
[2] Assistant professor, Dept. of Computer Sci., and Eng., Alvas Institute of Eng., Technol., Mangalore, India.

Abstract: In this work, we have presented an efficient approach for classification of text and non-text document information from real time office documents images printed/handwritten which are bilingual using a deep learning approach i.e., U-net architecture for experimentation purpose. We have created our own dataset containing 2000 document images. Initially pre-processing is applied on the input document images proposed method is compared with other existing methods and obtained accuracy of 99.62% different performance measure i.e., (Specificity, Sensitivity, Precision, F1-Score) used in the experimentation.
Keywords: Document Images; Pre-Processing; Filtering; Segmentation (U-Net).

1. Introduction

An imperative aspect of computer vision is the selection and classification of areas of interest in scanned images of text documents. Many researchers around the world are studying how to convert document images into editable formats. There needs to be a separation of text zones from non-text zones and a correct ordering of them in reading systems. An image can be analyzed to detect/extract/recognize text. For applications including optical character extraction, human-machine input distinction, spam detection, and machine-to-human input differentiation, text recognition and classification in natural images are very significant. Changes in the environment in which images are taken make it difficult for in-text recognition to recognize valuable full text in images. Image text detection identifies locations that contain meaningful whole text in an image. Taking an image in a different area makes it difficult. In analyzing document layouts, it is important to separate text and non-text elements.

The complex structure of the document has limited the quality of separation results despite several approaches. In order for the printed text to be recognized, it must be separated from non-text areas, such as signatures, handwritten text, logos, and other symbols, in order to be accurate. Most research, however, focuses on converting images of documents into the editable text because of the many ways in which this conversion can be used. Survey of text / non-text separation using various feature classifier combinations.

Documents written by hand are generally unstructured. They generally lack structure, i.e., they lack organization. Due to the lack of a specific layout, handwritten documents appear very chaotic compared to printed documents [4].

Data extraction and retrieval from digital documents have become nearly impossible with the rapid increase in digital documents. There is a need for automated methods. A variety of tools and methods are available to convert digital documents into text that can be processed. To understand and extract knowledge from documents, graphical elements like tables, figures, and equations are crucial. In the research community, therefore, the detection of these objects from documents has attracted considerable attention. Detecting tables and figures within documents is a challenging problem due to the lack of common dimensions and variations in their layouts. The purpose of this article is to recognize different types of digitally generated documents that contain graphical objects such as tables, diagrams, and equations. An object detection problem in a natural scene is conceptually similar to this problem. In rule- based systems, it is difficult to detect irregularly arranged equations, tables, and diagrams [1] – [8].

We present an end-to-end deep learning-based framework, called Visual Structure Object Recognition (VSOR), for detecting visual objects in document images, such as tables, figures, and

equations. Data- driven and independent of any heuristic rules for detecting visual objects in document images, our framework is based on recent object detection algorithms in computer vision [9] – [11]. As our task does not include labelled training data, deep learning-based methods require large amounts of data. VSOR explores transfer learning and domain adaptation in order to solve the scarcity of labelled training data in document images for Visual Structure Object detection/Recognition. The VSOR more accurately localizes all visual objects in document images than state-of-the-art techniques based on numerous public benchmark data sets.

Computer vision has experienced a revolution in object detection. This topic presents one of the most challenging aspects of computer vision due to its involvement in both object classification and object localization. A simple definition of this type of detection technique is to figure out where objects are located in a given image, known as object localization, and to which class of objects they belong, known as object classification. The accuracy of error calculation, accuracy, and recall are used to evaluate the performance of text recognition and localization systems. A variation in time or space complexity may result in poor results when character, word, language, script, etc., are misclassified and misinterpreted. The quality of the image and the representation of the image also affect these recognition techniques [5].

An image description framework based on neural networks. The CNN for image encoding can be replaced with an RNN encoding for the source text, based on the encoder-decoder model used in machine translation [6]. Using KCRAlexNet and KCRGoogLeNet [7], the structure of this describes the creation of a CNN network, starting from the organization of test data and ending with plotting the test accuracy curve.

A visual demonstration of how VSOR technique can successfully locate various visual objects in abstract images within a document. In particular, the contributions of this work are as follows:

The aim of this paper is to present an end-to-end trainable deep learning approach based on the concept of object detection algorithms used in recent years in computer vision to locate visual objects. To detect visual objects in document images, we refine a model trained by transfer learning. [9],[10], [11].

2. Related Work

In this section, we provide a brief overview of the literature regarding the detection and classification of texts and non-text. Tables, figures, equations, and other visual objects are necessary to detect and understand information from digital documents. The analysis and localization of different types of documents have been conducted by a number of researchers. An image can be recognized as text by identifying the areas where it is located. Many computer vision applications use text recognition and classification, such as optical character recognition, recognition of human and machine input, and spam detection [1].

Using a recursive filter to separate text from non-text in document layout analysis. Documents with complex background images can benefit from this method. During such a separation task, the author started with a length-normalized horizontal projection profile (HPP) [2], [3].

Document images using LBP-based features to separate handwritten texts from non-text images [4]. A method for recognizing bilingual machine printed images [5]. The task of image annotation encompasses both computer vision (CV) and natural language processing (NLP) [6]. Many areas of image recognition have been studied using convolutional neural networks (CNNs), and these studies have shown excellent results. KCR AlexNet, KCR GoogleNet, and CNN architecture performance [7].

Using a projection profile with variable threshold, the author presented an algorithm for segmenting text lines that increases accuracy and reduces computation time [8]. The classification of document images is important for Office automation, digital libraries, and other applications requiring document image analysis. Document image classifiers come in different types. Different methods use different methods for solving the problem, creating the class model, looking at the document's functionality, and choosing the classification algorithm [9]. XY trees represent documents. Hidden Markov models are used to classify documents with labels [10]. Analyze existing methods of classifying documents by focusing on

their image types. There are four types of methods: text-based, structure-based, image-based, and hybrid [11]. The purpose of this document is to describe functions and methods for comparing and classifying document images based on their spatial layout. Visual similarity searches and fast algorithms for identifying document types without OCR are useful methods for this type of search [12]. Segment the document into several patches using a simple region expansion algorithm. In NN and SVM models, labels are assigned to each segmented patch. It is possible to separate handwriting from machine printing with these models that support signatures as a type of handwriting. A comparison of SVM and NN showed that SVM had a superior classification rate. Based on the author's BoVW model, an optimal codebook construction is achieved through the use of the self-growth and self-organizing neural gas network [13– 14]. Based on Convolutional Neural Networks (CNN), a rapid one-dimensional approach to automated document layout analysis [15]. Any document analysis system must separate text from non-text [16]. Feature matching and nearest neighbor searches are used primarily for document search based on logo spot. The second method of feature matching is to use approximate nearest neighbors (ANN) [17]. A Neighbor classifier based on k-nearest neighbor verification and validation includes machine-printed text in different font sizes [18]. A variety of letter and word identification and recognition problems have been addressed using the TDE-PI approach [19]. The process of separating text from images in scanned documents is called text/image separation [20]. A variety of text classification techniques are analyzed in this paper. Text classification techniques are redesigned by artificial intelligence to better acquire knowledge [21]. A robust adaptive classifier is used to classify text documents [22]. There are several text classification algorithms discussed in this article, including text classification, text mining, text representation, text categorization, text analysis, and document classification [23]. A multifeatured AdaBoost classifier can be used to remove handwritten annotations or to more logically organize handwritten and printed text in mixed documents [24]. By identifying column and row line delimiters and their properties, the author explained how to detect table spaces in document images [25]. For effective implementation of data science technology, it is important to recreate tabular data from printed documents in digital format [26]. Using annotated machine-printed images from Bangla documents, the author described a method for distinguishing handwritten and machine-printed text. An SVM-based default classifier produces the final response by extracting Bengali character-specific features from this connected component image [27]. An ANN is more accurate if the K value and distance measurement are known for hierarchical classification. Deep learning algorithms are CNNs and RNNs [28]. In this article, the author proposed a new technique for automatic table recognition of document images. Rows and tables are one of the most common graphic non-text elements in a document, and their detection is directly related to OCR performance and document layout descriptions. Use horizontal and vertical line detection and table detection [29]. A comparison study of back propagation neural networks and support vector machines found that support vector machines are better at classification than back propagation neural networks [30]. Using FasterRCNN, the deep network that extracts these regions from images based on image transformations is based on FasterRCNN. Faster RCNNs combine regional proposal networks (RPNs) with high-speed RCNNs to form highly combination-dependent A networks [31–32]. By representing logical or quantitative relationships between information, a table structure adds another dimension to raw textual data. Graph Neural Networks (GNN) for recognizing invoice tables [33]. The author provides a set of rules for detecting equation places for Tesseract OCR [34]. A Deep Structured Prediction Based Supervised Clustering Approach for Detecting Page Objects from PDF Document Images [35]. According to this paper, a region-based feature extraction method determines whether the input region contains text or not by computing its RILTP features. In order to perform optical character recognition (OCR), it is imperative to separate text from non-text. Documents can be presented easily in the following formats after feature selection and conversion: ML. Text classifiers, such as probabilistic methods based on machine learning, are widely presented in the

literature. Various approaches are often used as the basis of assumptions: decision trees, naive bayes, rules guidance, neural networks, nearest neighbors, and it recently added support vector machines as well.

The automatic text classifier is not perfect and needs improvement, despite the many approaches for which an approach is proposed. [36, 37, 38]. Using deep learning, the author proposes a new framework for localizing graphical objects (GOD) in document images. The DeepDeSRT method demonstrates the ability to detect and recognize tables in documents using deep learning. [39, 40, 41]. There were several approaches developed before deep learning [42]-[45]. Computer vision tasks that require identifying visually distinguishable features in images have been made easier with Deep Convolutional Neural Networks (DCNNs). A number of research groups have studied the effectiveness of DCNN in analyzing documentary data in recent years. [46]-[49]. Table detection in PDF documents using deep learning. By using CNN, they categorize regions in a document with a table-like structure based on heuristics. As a result, this method is not entirely capable of overcoming the limitations of defining tabular structures based on rules [47]. Deep learning has been proposed for detecting and determining the structure of tables without making any assumptions about them. The authors refined Faster RCNN [9] using two backend architectures: ZFNet [53] and VGG16 [54]. A similar approach is also used by Gilani et al. [48]. A Faster RCNN model, however, is capable of only identifying table regions in the documents since it superimposes the three transformed layers at different distances. Multiple methods have been developed to identify text and non-text or graphical sections in documents [55]–[59]. A classification method can be classified based on three criteria: the first is region-based, the second is pixel-based, and the third is connected component-based. Due to the fact that graphical regions do not follow a universal pattern, none of these algorithms can identify graphical regions in document images. According to [49], each pixel could be classified into multiple classes using a saliency-based CNN architecture. Initially, this model retrieved saliency features concerning text, table, and figure sections. Classification maps are extracted from the saliency detector, and then binary classifiers are used to classify the segments. A binary classifier was trained separately for each classification in this model. This model can detect page objects such as tables, bar charts, pie charts, and line charts. An approach based on dynamic programming was suggested for the detection of page objects in [61]. A proposed algorithm for recognizing visual structure objects is described in this section, which is inspired by contemporary algorithms which identify objects in computer vision [50], [52]. A document image's graphical components, such as tables, figures, equations, and other objects, are located by detecting them. As with natural scene photos, this challenge involves detecting items. A CNN can be used to retrieve the visual characteristics of objects in natural scenes. CNNs can also identify graphical items based on certain properties. On PASCAL VOC [62] and COCO [63], Faster R-CNN and Mask RCNN were able to achieve compelling results, which motivate our effort. Our document object detection area lacks the data needed to train deep networks. Thus, we consider domain adaptation and transfer learning in our research. In this study, we attempted to detect graphical items in document images using two RCNNs designed for natural scenes. In the GOD model, pre-trained ImageNet [64] models are the backbone, which is derived from R-CNNs that run faster. There are two parts to Faster-RCNN. In the first module, the Region Proposal Network (RPN) suggests regions that could contain items. In the second module, the Fast-RCNN detector is implemented [65] and [66]. The input image is first processed by a convolution layer, which produces a feature map. In the case of Faster R-CNN, the Region of Interest (RoI) pooling layer uses max pooling to convert non-uniform size inputs to fixed size feature maps. For predicting item bounding box and class label, the output feature vector is sent via fully linked layers: box-regression (reg) and classification (cls). In the case of Mask R- CNN, the RoI Align layer generates fixed-sized features by preserving the exact spatial locations, whereas in the case of Mask R-CNN, the RoIAlign layer generates fixed-sized features by preserving the exact spatial locations. Finally, the fixed-sized features are passed to two separate modules: a multiple layer perceptron for predicting object

bounding box and a class label and mask module for predicting classification mask.

3. Proposed Method

In this section, we discuss our approach in detail. Figure 1 shows a flowchart of the proposed method. The proposed method mainly consists of three different stages they are pre-processing, Data augmentation, classification for experimentation purpose we have considered real time bilingual printed (Kannada and English scripts document images. the input images containing both text and non-text (here we have considered signature & Logo) information. If the input image contains graphs and tables, the efficiency will be reduced because the proposed algorithm will not be rained for graph, tables. Since the input images real time documents, may be blurred, noisy and some distortions may present. Performance may undergo if we process the documents without removing these noises. As a result, in order to improve performance, we must improve the documents by the use of some pre-processing techniques. In the subsequent sections, we discuss the Pre-processing, Data augmentation and Classification.

This influences improvement while training the network and the pre-processed yield would then be farmed into classification. Whenever a digital input image is divided into different subgroups to improve by decreasing complexity and make analysing simple and easy. Here, the deep constitutional neural network U-Net is used.

In this works documents all enhanced using spatial domain methods, Frequency domain methods (DFT) and Fuzzy approach, better enhancement in achieved.

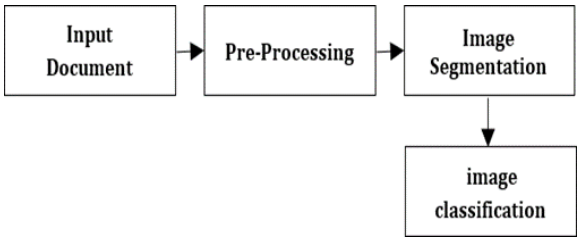


Fig. 1: Block diagram of proposed method.

3.1 Classification

U-Net classification uses document images in a CNN (convolutional neural network) structure for image classification. Data have outperformed an earlier satisfactory approach (a sliding-window convolutional network) for segmenting axonal frameworks in electron microscopic layers. U-Net structures are designed to contract and expand. The path of contracting is designed using CNNs. The convolutions are made up of 3x3 (unpadded convolutions) and can be repeated with rectified linear units (ReLu), along with a 2x2 pooling operation with stride 2. The number of function channels doubles with each down sampling step. In the course design, convolution kernels are used. The convolutions (unpadded convolutions) are repeated and observed via rectified linear units (ReLUs) with a 2x2 maximum pooling operation for down sampling. Each step of down sampling multiplies a number of channels examined. Expansive route starts with an up sampling of the feature space, then a convolution ("up-convolution") that cuts the range of characteristic channels in half, a concatenation with the consequently cropped function map from the contractual route, and two 3x3 convolutions with ReLUs. Convolutional networks lack boundary pixels, so cropping of the image is necessary. Finally, every 64-component feature vector is mapped to an apparent magnificence label using a 1x1 convolution. It appears that this state contains 23 convolutional layers. The input image is passed through the model by a convolutional layer with a ReLU activation function. In this case, we can see a decrease in image size from 2480X3508 to 1242X1754.Due to the use of unpadded convolutions to define the convolution layer as valid, the overall dimensionality was reduced. Additionally, there are encoder and decoder blocks on the left and right of the Convolution blocks.

Fig. 2. shows an architecture with encoders and decoders.

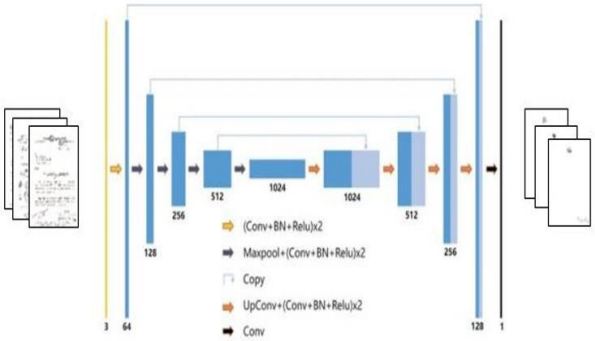
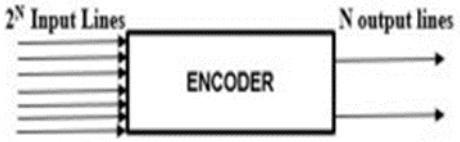


Fig. 2: U-Net architecture.

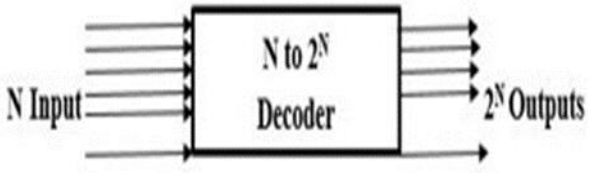
3.2 ENCODED PATH

Convolution layers are composed of 3x3 kernels, 2x2 Maxpool layers, and RELU activation functions. Consequently, this reduces the feature map's dimensionality, allowing hidden layers to remain and not just the most significant ones. Connections between U-nets are presented at the best layers, which reduces the wide range of parameters. By converting volatile statistics signals into coded messages, or analog warnings into virtual indicators, an encoder converts statistics signals into coded messages. An N bit code is represented by the N output lines resulting from the conversion of binary information into 2N input traces. The encoder converts a signal into coded binary output when it receives an input signal.



3.3 DECODED PATH

A segmented mask is made out of the input image after characteristic extraction in the encoding path. Decoders and encoders switch pooling indices during this step. The characteristic maps similar to the encoding course are copied to the decoding course. A decoder is a combinational circuit similar to an encoder, however, it operates in the opposite section. A decoder is a device that converts n traces of entering into 2n lines of output and generates the distinct sign as output from the coded input sign. Despite the excessive output produced by means of an AND gate, the primary interpreting element is that it produces a high output if all inputs are immoderate.



4. Result and Discussion

The U-Net method is used to separate the non-text from the document image. The obtained output is compared to the ground truth of the respective office document images to determine classification outputs. We imported the Unet model ResNet as the backbone network and loaded the image mesh weights. The output is passed to the U-Net model after it describes the layout of the input intended by the base model and indeed the specially designed overlay that obtains its base mode input. The UNet model's output is then propagated to other predefined ReLU-enabled ConvNet layers. The final result is reshaped to 1242x1754. Finally, we used the base model to construct a design that takes an input (x_inp) and outputs an output (x_out).

We defined the metrics, losses, and optimizer functions after compiling the model and defining everything that fitted the training and validation data to the proposed model. After saving the model, I used the trained model to create and save the X_train and X_test predictions. After making the predictions, we defined a function that visualises the model's predictions. This function expects input and output arrays as well as predictions. We obtained a mask for same dataset of the selected training sample by randomly selecting images from the training data and defining k as zero. Then I set the figure's size and plotted all three aspects: the image, the mask, and the predictive mask. The proposed approach yielded the following results, with the ground truth and predicted output for ideal office document images shown in Fig. 3.

Figure 4 shows the resultant output images 1–9, as well as the corresponding ground truth images and predicted outputs. (a) Represents input images, (b) Represents ground truth images and (c) Represents predicted images.

Following training, the performance of a machine learning classifier is evaluated using key performance metrics. The confusion matrix, which is a table showing whether a classifier needs to perform if some truth values/interests are gained, is among the performance metrics.

The most common matrices used for evaluating this architecture are accuracy, (F1) score, Precision, Sensitivity and Specificity. The proportion of classified instances pixels in an image.

$$\text{F1 Score, F1} = 2 * \frac{P_C * R_C}{P_C + R_C} \quad (1)$$

$$\text{Precision, Pc} = \frac{\text{True positive}}{\text{Truepositive} + \text{Falsepositive}} \quad (2)$$

$$\text{Sensitivity (Recall), Rc} = \frac{\text{Truepositive}}{\text{Truepositive} + \text{Falsenegetive}} \quad (3)$$

$$\text{Specificity} = \frac{\text{True negetive}}{\text{True negetive} + \text{False positive}} \quad (4)$$

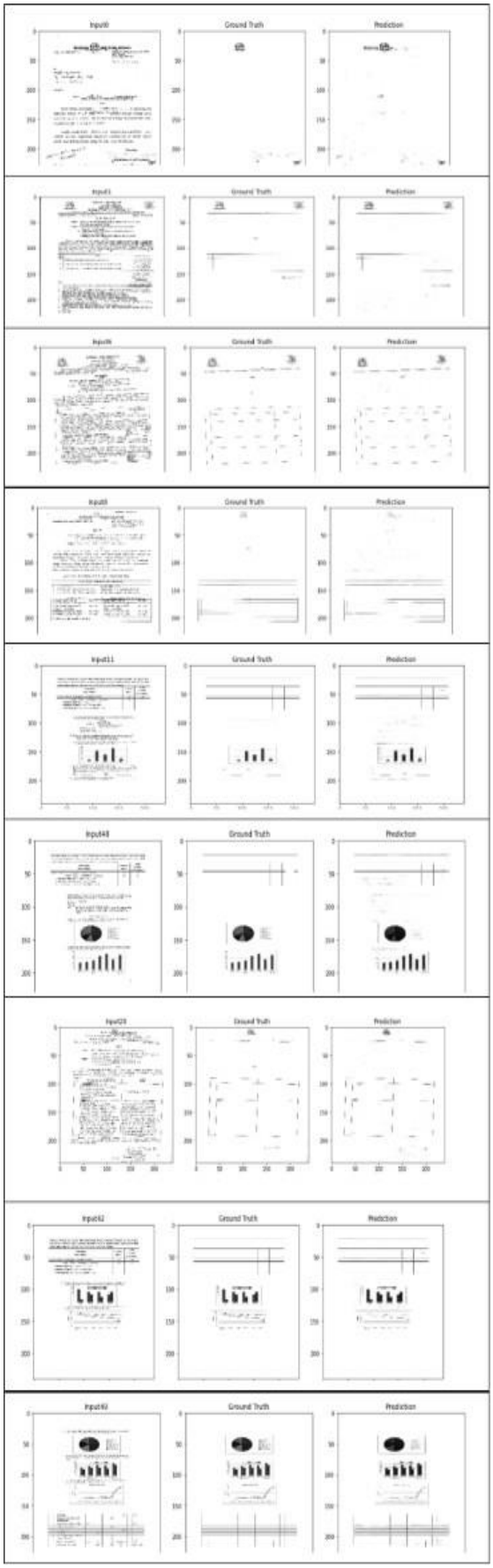


Fig. 3. Shows the resultant output images 1–9.

Table 1. Classification ROC FPR and TPR

FPR	TPR
0.1	0.2
0.2	0.5
0.3	0.9
0.4	1
0.5	1
0.6	1
0.7	1
0.8	1
0.9	1
1	1

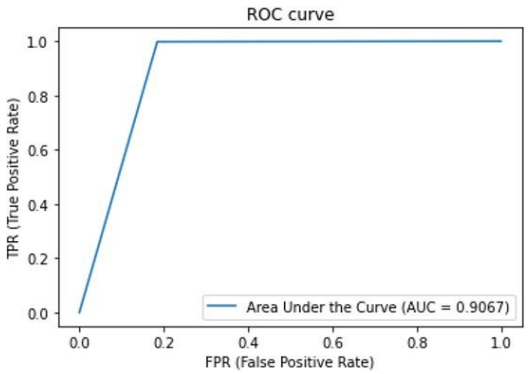


Fig. 4 ROC diagrams of the proposed U-Net for Real-time dataset.

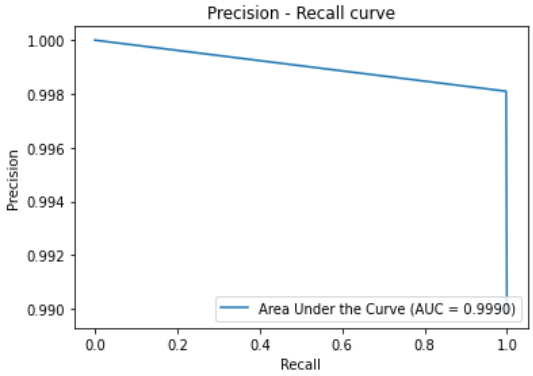


Fig. 5 Precision and Recall diagram of the present work U-net Classification.

```
Confusion matrix:  Custom threshold (for positive) of 0.5
[[  38284    8674]
 [   8525 4552517]]
Global Accuracy: 0.996267578125
Specificity: 0.8152817411303718
Sensitivity: 0.9981309095597015
Precision: 0.9980983037105878

F1 score (F-measure): 0.9981146063688572
```

The values of FPR and TPR for segmentation are given in table.1, and the ROC diagram of the proposed method is plotted in Fig. 4.

Precision and Recall diagram for the proposed method is plotted in Fig. 5. The accuracy loss diagram of the proposed method is plotted in Fig. 6, and the final diagram is plotted in Fig. 7.

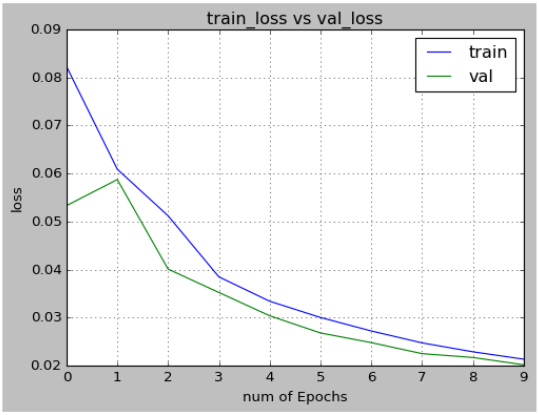


Fig. 6 The diagram of accuracy loss for training dataset.

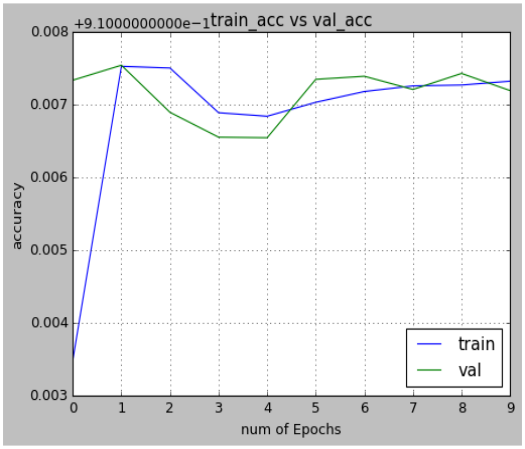


Fig. 7 the validation diagrams of the train & testing for Real-time dataset.

Table 2 Segmentation Accuracy of different.

Classification Methods	Accuracy
BiLSTM	95.65%
BiGRU	90.18%
CNN	93.42%
U-Net (Proposed)	99.62%

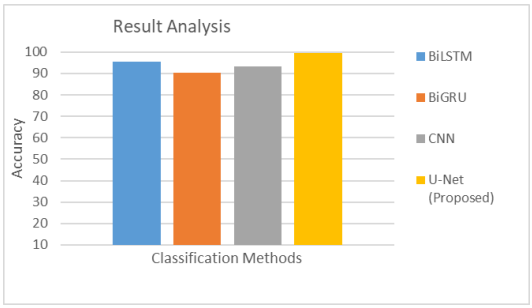


Fig. 8 Result analysis of proposed model.

Various segmentation accuracy methods are compared with the conventional method, and the results are presented in a table with a graphical representation in Fig. 8.

5. Conclusion

We proposed a deep learning approach, U-net architecture, in this paper to Classified non-textual data from bilingual office document images. Experimental work is accomplished out on our own dataset, and the results show that effectiveness of the proposed technique from the results we obtained accuracy of 99%, specificity of 64%, sensitivity of 99%, precision of 99% and F1-Score of 99%.

References

[1] C.P. Chaithanya, N. Manohar, Ajay Bazil Issac, Automatic Text Detection and Classification in Natural Images, International Journal of Recent Technology and Engineering (IJRTE), Volume-7, Issue-5S3, pp. 176-180, 2019.

[2] Tran, T. A., Na, I. S., & Kim, S. H. (2015). Separation of text and non-text in document layout analysis using a recursive filter. KSII Transactions on Internet and Information Systems (TIIS), 9(10), 4072- 4091.

[3] Arvind, K. R., Pati, P. B., & Ramakrishnan, A. G. (2006). Automatic text block separation in document images. In 2006 Fourth International Conference on Intelligent Sensing and Information Processing (pp. 53-58). IEEE.

[4] Ghosh, S., Lahiri, D., Bhowmik, S., Kavallieratou, E., & Sarkar, R. (2018). Text/non-text separation from handwritten document images using LBP based features: An empirical study. Journal of Imaging, 4(4), 57. pp. 01-15.

[5] Puri, S., & Singh, S. P. (2016, January). Text recognition in bilingual machine printed image

documents—Challenges and survey: A review on principal and crucial concerns of text extraction in bilingual printed images. In 2016 10th International Conference on Intelligent Systems and Control (ISCO) (pp. 1-8). IEEE.

[6] He, S., & Lu, Y. (2019). A Modularized Architecture of Multi-Branch Convolutional Neural Network for Image Captioning. Electronics, 8(12), 1417. Pp.01-15.

[7] Lee, S. G., Sung, Y., Kim, Y. G., & Cha, E. Y. (2018). Variations of AlexNet and GoogLeNet to improve Korean character recognition performance. Journal of Information Processing Systems, 14(1), 205- 217.

[8] Mishra, Shashank & Malathi, D. & Senthilkumar, Kavitha. (2018)., Digit Recognition Using Deep Learning, International Journal of Pure and Applied Mathematics, Volume 118 No. 22 2018, pp.295- 302.

[9] Chen, N., & Blostein, D. (2007). A survey of document image classification: problem statement, classifier architecture and performance evaluation. International Journal of Document Analysis and Recognition (IJDAR), 10(1), pp. 1-16.

[10] Diligenti, M., Frasconi, P., & Gori, M. (2003). Hidden tree Markov models for document image classification. IEEE Transactions on pattern analysis and machine intelligence, 25(4), 519-523.

[11] Liu, L., Wang, Z., Qiu, T., Chen, Q., Lu, Y., & Suen, C. Y. (2021). Document image classification: Progress over two decades. Neurocomputing, 453, pp.223-240.

[12] Hu, J., Kashi, R., & Wilfong, G. (1999, September). Document image layout comparison and classification. In Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR'99 (Cat. No. PR00318) (pp. 285-288). IEEE.

[13] Shirdhonkar, M. S., & Kokare, M. B. (2010). Discrimination between printed and handwritten text in documents. IJCA Special Issue on. Recent Trends in Image Processing and Pattern Recognition, pp.131-134, 2010.

[14] Zagoris, K., Pratikakis, I., Antonacopoulos, A., Gatos, B., & Papamarkos, N. (2014). Distinction between handwritten and machine-printed text based on the bag of visual words model. Pattern Recognition, 47(3), 1051-1062.

[15] Augusto Borges Oliveira, D., & Palhares Viana, M. (2017). Fast CNN-based document layout analysis. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 1173-1180).

[16] Bhowmik, S., Sarkar, R., Nasipuri, M., & Doermann, D. (2018). Text and non-text separation in offline document images: a survey. International Journal on Document Analysis and Recognition (IJDAR), 21(1), 1-20.

[17] Le, V. P., Nayef, N., Visani, M., & Ogier, J. M. (2016, March). Time-efficient Logo Spotting using Text/Non-text Separation as Preprocessing and Approximate Nearest Neighbor Search. In Semaine du Document Numérique et de la Recherche d'Information SDNRI 2016 (CORIA-CIFED) (pp. 365-380).

[18] Dhandra, B. V., Soma, S., Rashmi, T., & Gururaj, M. (2010). Classification of Document Image Components. International Journal of Engineering Research and Technology, 2(10), 1429-1439.

[19] Saxena, N., & Parveen, H. (2019). Text extraction systems for printed images: a review. International Journal of Advanced Studies of Scientific Research, 4(2). Pp.513-519, 2019.

[20] Kumar, S. S., Rajendran, P., Prabaharan, P., & Soman, K. P. (2016). Text/image region separation for document layout detection of old document images using non-linear diffusion and level set. Procedia Computer Science, 93, 469-477.

[21] Thangaraj, M., & Sivakami, M. (2018). Text classification techniques: a literature review. Interdisciplinary Journal of Information, Knowledge, and Management, 13, 117.

[22] Blessie, E.C., Deepa A, (2019). Classification of Text Documents Using Adaptive Robust Classifier. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-7, Issue-6, March 2019.

[23] Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. Information, 10(4), 150., pp.01-68.

[24] Lin, Y., Song, Y., Li, Y., Wang, F., & He, K. (2017). Multilingual corpus construction based on printed and handwritten character separation. Multimedia Tools and Applications, 76(3), 4123-4139.

[25] Kasar, T., Barlas, P., Adam, S., Chatelain, C., & Paquet, T. (2013, August). Learning to detect tables in scanned document images using line information. In 2013 12th International Conference on Document Analysis and Recognition (pp. 1185-1189). IEEE.

[26] Gupta, A., Tiwari, D., Khurana, T., & Das, S. (2019). Table detection and metadata extraction in document images. In Smart Innovations in Communication and Computational Sciences (pp. 361- 372). Springer, Singapore.

[27] Banerjee, P., & Chaudhuri, B. B. (2012, September). A system for handwritten and machine-printed text separation in Bangla document images. In 2012 International Conference on Frontiers in Handwriting Recognition (pp. 758-762). IEEE.

[28] Bhavani, A., & Kumar, B. S. (2021, April). A Review of State Art of Text Classification Algorithms. In 2021 5th International Conference on Computing Methodologies and Communication (ICCMC) (pp. 1484-1490). IEEE.

[29] Gatos, B., Danatsas, D., Pratikakis, I., & Perantonis, S. J. (2005, August). Automatic table detection in document images. In International Conference on Pattern Recognition and Image Analysis (pp. 609- 618). Springer, Berlin, Heidelberg.

[30] Ibrahim, Z., Isa, D., & Rajkumar, R. (2008, December). Text and non-text segmentation and classification from document images. In 2008 International Conference on Computer Science and Software Engineering (Vol. 1, pp. 973-976). IEEE.

[31] Gilani, A., Qasim, S. R., Malik, I., & Shafait, F. (2017, November). Table detection using deep learning. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 771-776). IEEE.

[32] Bavdekar, S. B. (2015). Using tables and graphs for reporting data. J Assoc Physicians India, 63(10), 59-63.

[33] Riba, P., Dutta, A., Goldmann, L., Fornés, A., Ramos, O., & Lladós, J. (2019, September). Table detection in invoice documents by graph neural networks. In 2019 International Conference on Document Analysis and Recognition (ICDAR) (pp. 122-127). IEEE.

[34] Liu, Z., & Smith, R. (2013, August). A simple equation region detector for printed document images in tesseract. In 2013 12th International Conference on Document Analysis and Recognition (pp. 245- 249). IEEE.

[35] Li, X. H., Yin, F., & Liu, C. L. (2018, August). Page object detection from pdf document images by deep structured prediction and supervised clustering. In 2018 24th International Conference on Pattern Recognition (ICPR) (pp. 3627-3632). IEEE.

[36] Ghosh, S., Hassan, S. K., Khan, A. H., Manna, A., Bhowmik, S., & Sarkar, R. (2022). Application of texture-based features for text non-text classification in printed document images with novel feature selection algorithm. Soft Computing, 26(2), 891-909.

[37] Julca-Aguilar, F. D., Maia, A. L., & Hirata, N. S. (2017, October). Text/non-text classification of connected components in document images. In 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI) (pp. 450-455). IEEE.

[38] Ikonomakis, M., Kotsiantis, S., & Tampakas, V. (2005). Text classification using machine learning techniques. WSEAS transactions on computers, 4(8), 966-974.

[39] Saha, R., Mondal, A., & Jawahar, C. V. (2019, September). Graphical object detection in document images. In 2019 International Conference on Document Analysis and Recognition (ICDAR) (pp. 51- 58). IEEE.

[40] Schreiber, S., Agne, S., Wolf, I., Dengel, A., & Ahmed, S. (2017, November). Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 1162-1167). IEEE.

[41] Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems, 30(11), 3212-3232.

[42] Chen, J., & Lopresti, D. (2011, September). Table detection in noisy off-line handwritten documents. In 2011 International Conference on Document Analysis and Recognition (pp. 399-403). IEEE.

[43] Tupaj, S., Shi, Z., Chang, C. H., & Alam, H. (1996). Extracting tabular information from text files. EECS Department, Tufts University, Medford, USA, 1.

[44] Fang, J., Gao, L., Bai, K., Qiu, R., Tao, X., & Tang, Z. (2011, September). A table detection method for multipage pdf documents via visual seperators and tabular structures. In 2011 International Conference on Document Analysis and Recognition (pp. 779-783). IEEE.

[45] Shafait, F., & Smith, R. (2010, June). Table detection in heterogeneous documents. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (pp. 65-72).

[46] Hao, L., Gao, L., Yi, X., & Tang, Z. (2016, April). A table detection method for pdf documents based on convolutional neural networks. In 2016 12th IAPR Workshop on Document Analysis Systems (DAS) (pp. 287-292). IEEE.

[47] Schreiber, S., Agne, S., Wolf, I., Dengel, A., & Ahmed, S. (2017, November). Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 1162-1167). IEEE.

[48] Gilani, A., Qasim, S. R., Malik, I., & Shafait, F. (2017, November). Table detection using deep learning. In 2017 14th IAPR international conference on document analysis and recognition (ICDAR) (Vol. 1, pp. 771-776). IEEE.

[49] Kavasidis, I., Palazzo, S., Spampinato, C., Pino, C., Giordano, D., Giuffrida, D., & Messina, P. (2018). A saliency-based convolutional neural network for table and chart detection in digitized documents. arXiv preprint arXiv:1804.06236.

[50] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.

[51] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).

[52] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).

[53] Zeiler, M. D., & Fergus, R. (2014, September). Visualizing and understanding convolutional networks. In European conference on computer vision (pp. 818-833). Springer, Cham.

[54] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[55] Okun, O., Dœrmann, D., & Pietikainen, M. (1999). Page segmentation and zone classification: the state of the art. Pp.01-34.

[56] Moll, M. A., & Baird, H. S. (2008, January). Segmentation-based retrieval of document images from diverse collections. In Document Recognition and Retrieval XV (Vol. 6815, p. 68150L). International Society for Optics and Photonics.

[57] Nayef, N., & Ogier, J. M. (2015, August). Text zone classification using unsupervised feature learning. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR) (pp. 776- 780). IEEE.

[58] Fletcher, L. A., & Kasturi, R. (1988). A robust algorithm for text string separation from mixed text/graphics images. IEEE transactions on pattern analysis and machine intelligence, 10(6), 910- 918.

[59] Tombre, K., Tabbone, S., Pélissier, L., Lamiroy, B., & Dosch, P. (2002, August). Text/graphics separation revisited. In International Workshop on Document Analysis Systems (pp. 200-211). Springer, Berlin, Heidelberg.

[60] Le, V. P., Nayef, N., Visani, M., Ogier, J. M., & De Tran, C. (2015, August). Text and non-text segmentation based on connected component features. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR) (pp. 1096-1100). IEEE.

[61] Yi, X., Gao, L., Liao, Y., Zhang, X., Liu, R., & Jiang, Z. (2017, November). CNN based page object detection in document images. In 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR) (Vol. 1, pp. 230-235). IEEE.

[62] Everingham, M., Van Gool, L., Williams, C. K., Winn, J., & Zisserman, A. (2010). The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2), 303-338.

[63] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.

[64] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255). Ieee.

[65] Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).

[66] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).