# IDENTIFICATION OF TEXTUAL REGION OF INTEREST FROM COMPLEX IMAGES

**[1]Sumathi koralapati, [2]Dr. Gnana Prakash T**
[1]M-Tech Student , Dept of Computer Science, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Telangana, India. 500090
Sumakorlapati703@gmail.com
[2]Professor Dept of Computer Science, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Telangana, India.500090
gnanaprakash_t@vnrvjiet.in

**Abstract-**

The identification of textual regions of interest (ROI) in images plays a crucial role in various applications, including document analysis, image understanding, and optical character recognition (OCR). This research focuses on developing effective methodologies for automatically detecting and extracting textual content from diverse images. The proposed approach leverages advanced computer vision and machine learning techniques to robustly identify and isolate textual regions within images. Initially, image preprocessing techniques are employed to enhance the quality and clarity of the input images. Subsequently, a combination of feature extraction and deep learning algorithms is applied to analyze the visual patterns associated with text. The model is trained on a diverse dataset to ensure its adaptability to various types of textual content, fonts, and backgrounds. Transfer learning techniques are explored to enhance the model's generalization capabilities, allowing it to perform well on unseen data. The training process involves optimizing the model parameters to achieve high accuracy and efficiency in ROI identification. To evaluate the proposed methodology, extensive experiments are conducted on benchmark datasets, comparing the performance against state-of-the-art methods. The results demonstrate the effectiveness of the proposed approach in accurately identifying textual ROIs across different scenarios, including complex backgrounds, varying fonts, and diverse image resolutions.

Furthermore, the research explores potential applications of the identified textual ROIs, such as improving OCR systems, document summarization, and information retrieval. The robustness and adaptability of the proposed methodology position it as a valuable tool for text-related tasks in real-world applications. In conclusion, the developed approach showcases significant advancements in the identification of textual regions of interest in images, contributing to the broader field of computer vision and image processing. The potential impact of this research extends to fields such as automation, document analysis, and intelligent information extraction, fostering advancements in technology and enhancing the efficiency of various applications.

**Keywords:** East methodology, Mser, Moron methodology Craft method, OCR

**DOINumber:10.48047/nq.2023.21.7.nq23081** **NeuroQuantology2023;21(7):925-935**

**INTRODUCTION TO OCR**

(Optical Character Recognition)

The challenge of extracting text from images of documents has traditionally been referred to as Optical Character Recognition (OCR) and has been the focus of much research. When documents are clearly laid out and have global structure (for example, a business letter), existing tools for OCR can perform quite well. A popular open source tool for OCR is the Tesseract Project, which was originally developed by Hewlett-Packard but has been under the care and feeding of Google in recent years. Tesseract provides an easy-to-use interface as well as an accompanying Python client library, and tends to be a go-to tool for OCR-related projects. More recently, cloud service providers are rolling out text detection capabilities alongside their various computer vision offerings. These include Google Vision, AWS Textract, Azure OCR, and Drop box, among others. It is an exciting time in the field, as computer vision techniques are becoming widely available to empower many use cases.

There are, however, many use cases in what we might call non-traditional OCR where these existing generic solutions are not quite the right fit. An example might be in detecting arbitrary text from images of natural scenes. Problems of this nature are formalized in the COCO-Text challenge, where the goal is to extract text that might be included in road signs, house numbers, advertisements, and so on. Another area that poses similar challenges is in text extraction from images of complex documents. In contrast to documents with a global layout (such as a letter, a page from a book, a column from a newspaper), many types of documents are relatively unstructured in their layout and have text elements scattered throughout (such as receipts, forms, and invoices). Problems like this have been recently formalized in the ICDAR DeTEXT Text Extraction From Biomedical Literature Figures challenge. These images are characterized by complex arrangements of text bodies scattered throughout a document and surrounded by many "distraction" objects. In these images, a primary challenge lies in properly segmenting objects in an image to identify reasonable text blocks. Example images from COCO-Text and ICDAR-DeTEXT are shown below. These regimes of non-traditional OCR pose unique challenges, including background/object separation, multiple scales of object detection, coloration, text orientation, text length diversity, font diversity, distraction objects, and occlusions.

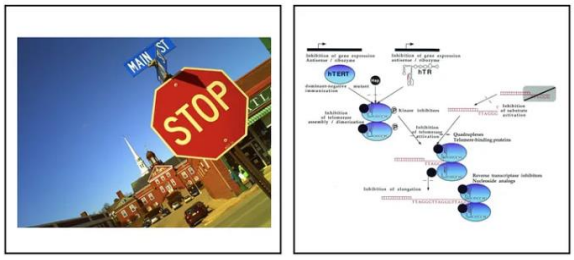



Figure 1. Example images from COCO-Text challenge (left) and ICDAR DeTEXT challenge (right). Notice that OCR in this regime requires the detection of text objects as separate from background pixels and other distractions

The problems posed in non-traditional OCR can be addressed with recent advances in computer vision, particularly within the field of object detection. As we discuss below, powerful methods from the object detection community can be easily adapted to the special case of OCR.

**OBJECT DETECTION**

The field of computer vision aims to extract semantic knowledge from digitized images by tackling challenges such as image classification, object detection, image segmentation, depth estimation, pose estimation, and more. For this discussion, we'll focus on the field of object detection (and related image segmentation) which has seen impressive improvements in recent years. Early attempts at object detection focused on applying image classification techniques to various pre-identified parts of an image. Many approaches have focused on speeding up the identification of candidate regions and on using convolutional mechanisms for feature extraction and classification. While there have been many interesting developments in the field, we will focus primarily on MaskRCNN, a model which is able to very successfully conduct object detection and image segmentation.

An example output from Mask RCNN is shown below. For any input image, this model is trying to accomplish three things: object detection (green boxes), object classification, and segmentation

(colorful shaded regions). The green bounding boxes on the image below are the outputs of the model and above each box is a prediction of what kind of object is contained within. We can see that when this scene of a busy street is fed into the model, Mask RCNN is able to successfully identify a large variety of different scene objects including people, cars, and traffic lights. Further, inside each identified bounding box, the colorful shaded region identifies exactly which pixels in an image correspond to the object. This is referred to as segmentation, and each pixel in the image receives a predicted classification label about what kind of object that pixel belongs to (or background).

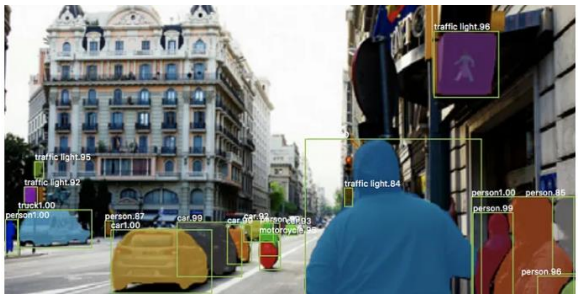



Figure 2: Output of Mask RCNN for an image of a busy street scene. Notice that relevant objects are detected (people, cars, traffic lights) with bounding boxes as well as segmentation regions.

Mask RCNN is an example of a multi-task network: with a single input (image), the model must predict multiple kinds of outputs. Specifically, Mask RCNN is split into three heads, where one of the heads is concerned with proposing bounding boxes that likely contain objects of interest, another head is concerned with classifying which type of object is contained within each box, and the final head identifies a pixel-wise bitmask for estimating the segmentation in each box. Importantly, all three of the heads are relying upon a shared representation that is calculated from a deep convolutional backbone model such as Res Net or similar. This shared representation is important in multi-task learning, and allows each of the heads to back-propagate their respective errors and update this backbone representation. The overall effect of this is that each head actually becomes more accurate than if they were trained as separate models.

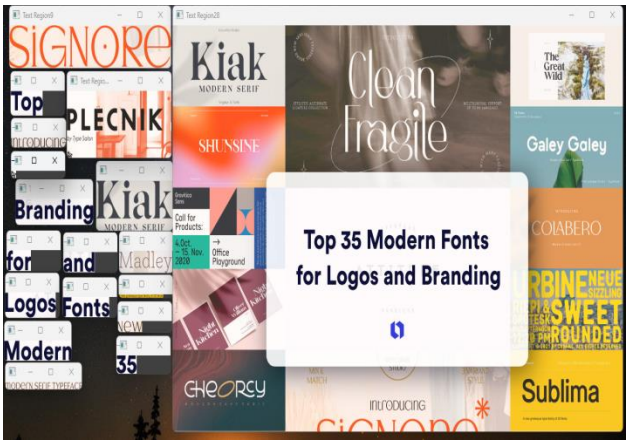




Figure 3: Output of Mask RCNN for an image

**Detecting Text Objects**

Object detection models such as Mask RCNN and its predecessors provide a very flexible mechanism for identifying regions of interest inside images. As you might guess, we can view non-traditional OCR as closely related to object detection. In this case, we have really only two classes of objects we care about: text objects and then everything else. With this perspective, we can train a model very similar to Mask RCNN to identify regions of interest (RoI) in an image that are highly likely to contain text, a task that is known as text localization. An example output of such a model is shown below.

Notice that this input image of a receipt presents some interesting challenges for text extraction. First, the document of interest occurs alongside some background objects (a steering wheel). Second, the text within the document is highly unstructured and therefore it is beneficial to separately identify all the possible text blocks. The output of the model is overlaid on the image above

— regions of text are identified with dotted-line bounding boxes and we even have estimated the pixel mask for the text. Above each box is the predicted class and confidence score, which, since we only have one object class of interest, is "Text" in all detected cases. Note that the bounding boxes are all quite tight and encapsulate the text regions fairly accurately. Modifying a model like Mask RCNN and training it with OCR-relevant datasets leads to an effective approach for text localization. It is of interest to note that even though the pixel masks are not inherently required for OCR, we have observed that including this constraint in the multi-task learning forces the localization (bounding box regression) to be even more accurate.

If all we can do is identify RoIs of an image that correspond to text blocks, then this is obviously of limited utility for OCR. But what we need to do next is read the text contained in each image region; this is known as text recognition. The model described below is a departure from Mask RCNN and is a multi-task network for solving both text localization and text recognition.

**Multi-Task Network for Text Extraction**

Taking inspiration from models like Mask RCNN, we have designed our own multi-task network for solving both text localization and text recognition. Similar to previous approaches, our model entails a convolutional backbone for extracting image features. We have evaluated both Res Net and Densely Connected Convolutional Networks (DenseNet), for which we find that DenseNet leads to higher accuracy. Additionally, the output of the convolutional stack is then input into a Feature Pyramid Network which helps to combine high spatial resolution information from early in the stack with low-resolution but rich semantic detail from deeper in the stack. These form the basis of the convolutional backbone which is then sent to the model heads.

Similar to recent object detection models, the text localization head is comprised of a two-stage mechanism with a Region Proposal Network followed by a Bounding Box Regression network. The output of the latter component is a set of predicted boxes (RoIs) that might contain text. The second head of the model is the classification component which is tasked with estimating the class of object contained inside of each RoI — in this case, a simple binary classification (text vs. background). Finally, we have the text recognition head which takes as input the feature maps from the convolutional backbone and the RoI coordinates generated from the text localization head. This text recognition head must, for each RoI, produce a predicted sequence corresponding to the text inside each box and uses a CTC loss for training.

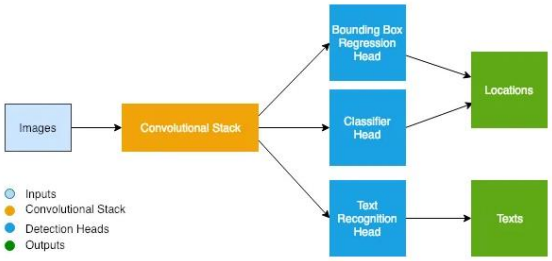



Figure 4: A simplified architecture of our multi-task model. Input images are passed through a convolutional stack which includes Res Net as well as Feature Pyramid Network. The resulting representations are then used by the model's multiple heads in order to identify text locations and sequences.

The text recognition head is a primary point of departure for our model as compared to object detection methods, so some additional detail is worthwhile. In object detection, feature maps are extracted from multiple levels of the convolutional backbone and are pooled to a fixed representation through a mechanism known as RoIPool (or RoIAlign). In OCR, we must retain high spatial resolution information, so we extract features only from early blocks of the convolutional backbone. Additionally, we rely on a new pooling mechanism that allows objects of different aspect ratios to be represented without compressing long sequences or stretching short ones.

Additionally, one is presented with vast possibilities for how to implement the text recognition head. A simple approach might be to crop the input image as identified by the RoIs from the Bound Box Regression head and then process this cropped image through an RNN architecture [16]. The limitation of such an approach is that we do not reuse image feature representations between the heads, which would require the recognition head to do more computation on its own. Instead, our multi-task network proceeds by using the identified

RoIs and then fetching the relevant representations for each region from the convolutional backbone. Each RoI's feature map is then transformed into a fixed shape as described above and the text recognition can proceed.

A typical approach for sequence classification and sequence labelling tasks would be to use an RNN architecture of some kind where we sequentially process from "left to right" across the RoI's approximately 200 horizontal spatial steps and try to predict an output label for each location in the RoI feature map. However, we find that RNNs perform quite poorly for text recognition here. This is likely due to the fact that we don't really need to account for long-range correlations in this sequence in order to decode the text. Instead, we only need to look at a few "feature columns" at a time in order to get a sense of which character is being represented. For this reason, we find it most useful to use simple convolutional methods here with short-range kernel widths. At each spatial step, the output of convolutions is used to predict an output letter, and then the overall sequence is collapsed through the CTC layer to output the final sequence for the ROI.

In order to train the model described here, we would need a large number of labeled images. In lieu of tagging and generating these manually, we instead chose to develop our own synthetic training documents. With enough variability in fonts, sizes, colors, distractions objects, and so on, our synthetic data should result in a model which is able to perform well on real-world images. We generated around ten thousand such images which led to strong performance on the real world cases we highlight here.

Example outputs of our model are shown below. On the left is one of the images from the ICDAR DeTEXT challenge and on the right is a screenshot of a receipt. Text segments are identified apart from background pixels and other image objects and are highlighted with dashed lines. The predicted text sequence for each RoI is shown in red above each box. Note that this model is able to accurately identify text object which occur with a large variety of aspect ratios, fonts, font sizes, and colors.

**Limitations of OCR technology**

More often than not, OCR software extracts data from image-based documents. However, the uptick in scanned documents with different formats, fonts, styles, and colors has led to several OCR limitations that include:

**1. Accuracy**

Generally, advanced OCR software has an accuracy rate of 99%, provided the input is a high-quality, black-and-white image with large fonts.

However, the accuracy rate is often compromised when the document processing software deals with handwritten content, intricate layouts, or skewed texts. OCR software also generates incorrect readings from minuscule texts and low-quality images. These inaccuracies affect the extracted data's overall quality and integrity.

**2. Language and font support**

The optical character resolution platform uses pattern recognition algorithms to match the scanned texts and characters with the ones present in its database. Naturally, the system generates inaccurate readings when it encounters fonts or languages that deviate from its pre-fixed parameters.

Since the algorithm is not adaptive, it may fail to identify unique language symbols or misinterpret certain characters. Consequently, organizations struggle to use OCR technology for effective multilingual and diverse document processing.



**3. Formatting errors**

Simple optical character recognition applications do not preserve the document's original formatting. Without third-party software aids, OCR technology struggles to process line breaks, font styles, indentations, tables, and graphs.

As a result, the newly generated document contains misaligned text, erroneous spacing, incorrect line breaks, and incomprehensible tables. These formatting errors are manually ironed out by the employees, decreasing productivity and adding unnecessary load to the workforce.

**4. Sensitive information exposure**

OCR software's parameter-based data extraction technique might inadvertently upload sensitive information, such as IDs, confidential documents, and financial data to the software provider's server. The data then becomes vulnerable to cyber attacks and security breaches. It becomes essential to mask and redact such information before putting the

documents through the scanner.

Furthermore, companies need to ensure proper storage and utilization of this sensitive information as per GDPR and SOC-2 guidelines. Any irresponsible handling of these confidential documents warrants harsh penalties. For instance, Meta (formerly Face book) was fined a staggering $1.3 billion for violating GDPR norms in Europe.

**5. Dependency on image quality**

OCR software's performance hinges on the quality of the source images or documents. Low-resolution images, faded text, or poor lighting conditions can introduce noise that hinders accurate character recognition. Blurry or distorted images might cause the software to misinterpret characters, leading to transcription errors and requiring manual intervention to rectify discrepancies.

**6. Time and resource intensive**

Integrating OCR software with your existing business stack is no easy task. It requires significant investments in time, capital, and human resources. The integration process starts with configuring the software to work seamlessly with your legacy systems.

Since it requires high-quality images to generate the best results, you would need to invest in ancillary applications to improve the input quality. If the organization is processing documents with a unique formatting style, additional capital will be required to develop the custom pattern recognition algorithm.

Lastly, the business needs to create a training module to teach the employees about the newly integrated system.

**7. Learning curve**

The OCR training module usually covers everything from pre-processing the images to ethical storage of the extracted information. However, gaining operational proficiency requires time. The adoption process of the newly implemented system might be slow as employees come to grips with the user interface, troubleshooting methods, and configuring parameters for pattern recognition.

Subsequently, the company also needs to account for the opportunity cost lost due to the steep learning curve and slow adoption rate of OCR software.

8. Lack of contextual and semantic understanding

Simple OCR software lacks any machine learning or natural language processing algorithms. Therefore, while it excels at character recognition, it fails to understand the nuances of a text passage and the relationships between the extracted data.

This limitation is most noticeable when companies try to create interconnected knowledge repositories using only OCR platforms. In doing so, they can only establish connections using keywords, not the search intent of the user.

9. Restricted application domains

Even though the OCR system is versatile, its limitations prevent widespread adoption in all major industries. For example, this system struggles with handwritten drawings, special equations, and complex scientific symbols.

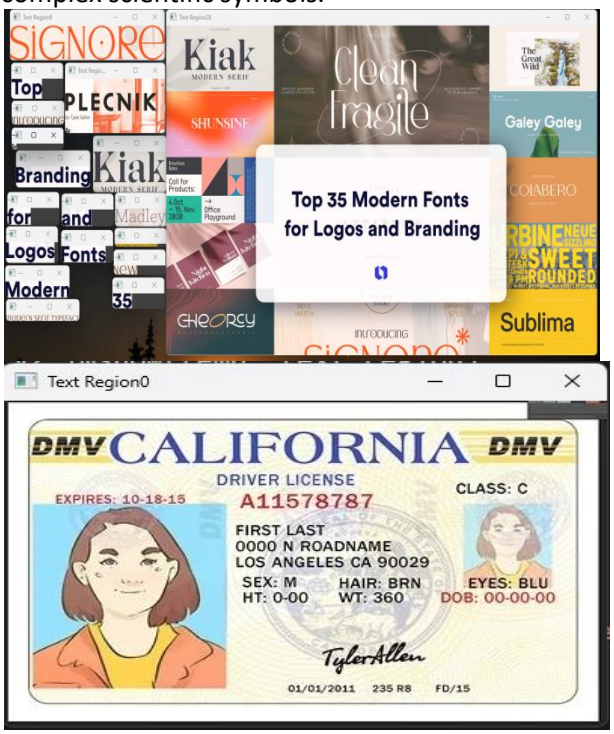



Figure 5: Output of Mask OCR for an image

**EAST text detector**

**Introduction**

Text Detection is a major problem in optical character recognition (OCR) and there are various solutions attempted by different researchers. EAST is another attempt made for detection of Scenic Text detection that's detection of text from images where background is normal street or billboard and model needs to detect text from that image. Later on, is realized that the same model can be very

useful in text detection from scanned images as well.

**EAST Model**

EAST is a deep learning-based algorithm (EAST) that detects text with a single neural network with the elimination of multi-stage approaches. The key component in this proposed algorithm is a neural network model, which is trained to directly predict the existence of text instances and their geometries from full images. Also, since this model has been designed as a fully-convolutional neural network adapted for text detection, it outputs dense per-pixel predictions of words or text lines thus eliminating the need of intermediate steps which were there in traditional models such as candidate proposal, text region formation and word partition.

The key highlights of EAST (Efficient and Accurate Scene Text Detector) are as follows:

• They propose a scene text detection method that consists of two stages: a Fully Convolutional Network and an NMS merging stage. The FCN directly produces text regions, excluding redundant and time-consuming intermediate steps.

• The pipeline is flexible to produce either word level or line level predictions, whose geometric shapes can be rotated boxes or quadrangles, depending on specific applications.

• The proposed algorithm significantly outperforms state-of-the-art methods in both accuracy and speed.

If you want to go deeper into EAST and learn how you can do training as well using ICDAR SIROE Dataset then you can learn it with Live Demo.

these models work and are implemented then Coding exercises with live examples can be accessed at Code Implementation of Object Detection using CNN.

**EAST Methodology**

The key component in this proposed algorithm is a neural network model, which is trained to directly predict the existence of text instances and their geometries from full images. Also, since this model has been designed as a fully-convolutional neural network adapted for text detection, it outputs dense per-pixel predictions of words or text lines thus eliminating the need of intermediate steps which were there in traditional models such as candidate proposal, text region formation and word partition. Even, the post-processing steps only include thresholding and NMS on predicted geometric shapes.

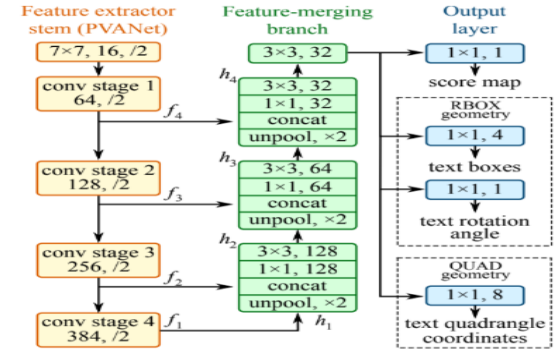



Figure 6: network architecture diagram

If we look at the EAST network architecture diagram on screen, it has 3 branches that combine into a single neural network.




Figure 7: Output of Mask EAST Methodology for an image

**These branches are:**

**Feature Extractor Stem –** It is used to extract features from different layers of the network. Within this, the stem can be a convolutional network pre-trained on ImageNet dataset, with interleaving convolution and pooling layers. As we can see in the diagram, four levels of feature maps, denoted as f1, f2, f3 and f4 are extracted from the

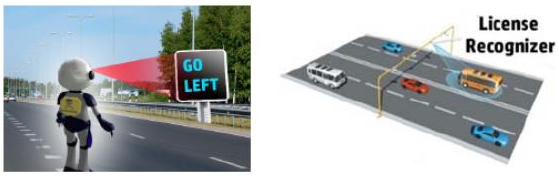
stem.

**Feature Merging Branch** – This branch of the EAST network merges the feature outputs from a different layer of the VGG16 network. The input image is passed through the VGG16 model and outputs from different four layers of VGG16 are taken and then feature maps are merged using U-net architecture. So, h1, h2, h3 and h4 are the merged feature maps.

**Output Layer** – The output layer consists of a score map and a geometry map. The score map tells us the probability of text in that region while the geometry map defines the boundary of the text box. This geometry map can be either a RBOX also called as rotated box or QUAD that is quadrangle. A rotated box consists of top-left coordinate, width, height and rotation angle for the text box. While quadrangle consists of all four coordinates of a rectangle.

**Mser Method**

Image contains many technical and digital information used in the different fields of computer vision. In recent years, visual detection and recognition of text from image is claimable because of its application in content based image searching, robotic navigation (Fig. 1a), automatic car number plate recognition (Fig. 1b), extracting passport or business card or bank statement information, converting handwriting to real time control of computer, making editable the text of any image etc. But owing to irregular background, variations of font style, size, color, orientation, geometric and photometric distortion, text must have to be robustly detected from any natural scene image Text detection has been considered in several current studies in various competitive methods. Existing methods of the detection of text can be classified into three groups: (1) texture based method [3]; (2) Connected Component (CC) based method and hybrid method [1]. In texture based method text is considered as exclusive texture that is dissimilar from the irregular background. A trained classifier is engaged on the job of identification of existence of text on the image. The segmented texture is filtered with nonlinear portion. The output of the filter at tanh(αt), where α = 0.25. If images are more complicated, texture segmentation scheme is not sufficient [6]. Connected component based method find out character candidates from image by connected component analysis which is followed by grouping character candidates to text.




(a) (b)

Figure 8. Applications of text detection and recognition. (a) Robotic navigation; (b) Automatic license plate checking system.

Supplementary checks have to be performed to remove false position. In hybrid method text is detected by extracting connected components as character candidates by binarization. Non characters are erased by Conditional Random Fields (CRFs). Recently, MSER based text detection has become a heart among all types of text detection method [2]. Maximally Stable Extremal Regions (MSER) works with the intensity management of a digital image. The term “extremal region” represents connected component which differentiate the higher or lower intensity of a pixel to the outer boundary pixel. Although MSER is actually in the family of connected component based method, different new projects have taken a great importance on it because of its promising performance. In this paper, we propose an enhanced MSER based detection combining with text recognition. Firstly MSER approaches is applied to a scene image to detect a large region of non-character which employs MSER as basic connected component. Since natural scene images are normally affected by a wide variation of background, to detect small text or to detect text from blur image or limited resolution image canny edge and MSER have to be combined. For obtaining more reliable result we have used stroke width transformation of the image to perform filtering and joining connected components. Finally, an Optical Character Recognition (OCR) technique with clustering by Gaussian Mixture Model (GMM) is operated on the selected text region to recognize the actual text information of the natural scene image. Our method exhibits outstanding

performance on natural scene detection and on ICDAR 2011 Robust Reading Competition dataset. The results rank on the first position for our best performance 77.47% in f-measure.

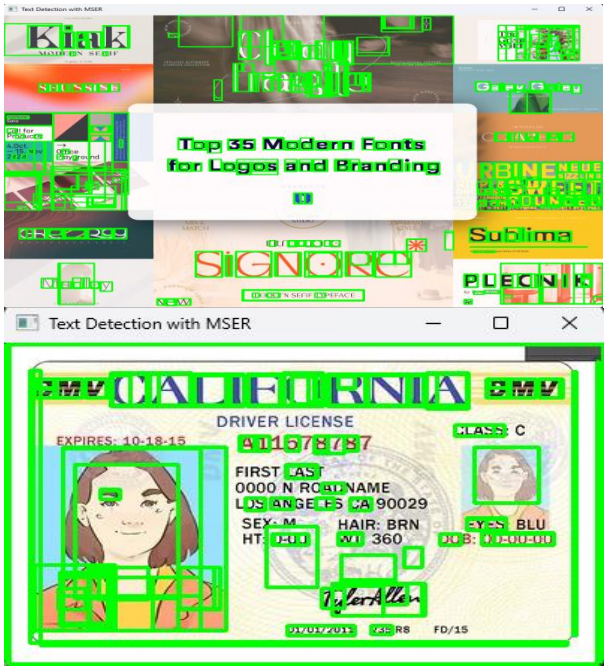



Figure 9: Output of Mask EAST Methodology for an image

**MMOCR**

In recent years, deep learning has achieved tremendous success in fundamental computer vision applications such as image recog-nition, object detection and image seg-mentation. In light of this, deep learning has also been applied to areas such as text detection and text recognition, as well as their downstream tasks such askey information extraction and named entity recognition. Dierent approaches utilize dierent training datasets, optimization strategies (e.g., optimizers, learning rate schedules, epoch num-bers, pre-trained weights, and data augmentation pipelines), andnetwork designs (e.g. network architectures and losses). To encom-pass the diversity of components used in various models, we haveproposed the MMOCR toolbox which covers recent popular textdetection, recognition and understanding approaches in a uni edframework. As of now, the toolbox implements seven text detec-tion methods, ve text recognition methods, one key information

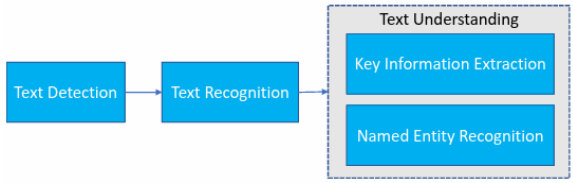



Figure 10: Overview of MMOCR. The supported text detection algorithms include DB, Mask R-CNN, PANet, PSENet, Text Snake, DRRG, and FCENet. The supported text recognition algorithms are CRNN, NRTR, Robust Scanner, SAR, and Seg OCR. The supported key information extractional gorithmis SDMG-R, and the supported named entity extraction algorithm is Bert-Softmax. method and one named entity recognition method. Integrating various algorithms confers code reusability and therefore dramatically simplifies the implementation of algorithms. Moreover, the unified framework allows different approaches to be compared against each other fairly and that their key effective components can be easily investigated. To the best of our knowledge, MMOCR reimplements the largest number of deep learning-based text detection and recognition approaches amongst various open-source toolboxes, and we believe it will facilitate future research on text detection, recognition and understanding.

Extracting structured information such as “shop name”, “shop address” and “total payment” in receipt images, and “name” and“organization name” in document images plays an important role in many practical scenarios. For example, in the case of office automation, such structured information is useful for efficient archiving or compliance checking. To provide a comprehensive pipeline for practical applications, MMOCR reimplements not only text detection and text recognition approaches, but also their downstream tasks such as key information extraction and named entity recognition as illustrated in Figure 1. In this way, MMOCR can meet the document image processing requirements in a one-stop-shopping manner.

MMOCR is publicly released at https://github.com/open-mmlab/mmocr under the Apache-2.0 License. The repository contains all the source code and detailed documentation including installation instructions, dataset preparation scripts, API documentation, model zoo, tutorials and user manual. MMOCR re-implements more than ten

state-of-the-art text detection, recognition, and understanding algorithms, and provides extensive benchmarks and models trained on popular academic datasets. To support multilingual OCR tasks,

MMOCR also releases Chinese text recognition models trained on industrial datasets 1. In addition to (distributed) training and testing scripts, MMOCR offers a rich set of utility tools covering visualization, demonstration and deployment. The models provided by

MMOCR are easily converted to onnx 2 which is widely supported by deployment frameworks and hardware devices. Therefore, it is useful for both academic researchers and industrial developers.

**CONCLUSION**

Identifying the textual region from the whole image makes the task of finding wanted text from the given complex images very easy and simple task. and in this project we are mainly working on the complex images which have many drawbacks ,working on the complex images is not an easy task, but using some technologies it can be implemented and this is mainly useful for the people who are physically impaired (blind people) identifying a text from any image has become very useful for today's generation which makes the work easier and smart .as an addon to this we are using east methodology which makes the task more easy and interest for finding the text from the complex images.

***REFERENCES***

[1] Jan, Mardhiyah & Zainal, Nasharuddin & Jamaludin, Shahrizan. (2020). Region of interest-based image retrieval techniques: a review. 9. 520. 10.11591/ijai.v9.i3.pp520-528.

[2] Prajwal. R Athrey, Asha K, Rajesha. H, Sahana. S Patil, Channabasavarajan. V, 2021, Region of Interest Extraction using Optical Character Recognition Template, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) ICACT – 2021 (Volume 09 – Issue 08),

[3] Selva Kumari, R.Shantha & Sangeetha, R.. (2015). Optical character recognition for document and newspaper. International Journal of Applied Engineering Research. 10. 15279-15285.

[4] Sonkusare, Manoj. (2017). A Study on Optical Character Recognition Techniques. International Journal of Computational Science, Information Technology and Control Engineering. 4. 01-15. 10.5121/ijcsitce.2017.4101.

[5] Hamad, Karez & Kaya, Mehmet. (2016). A Detailed Analysis of Optical Character Recognition Technology. International Journal of Applied Mathematics, Electronics and Computers. 4. 244-244. 10.18100/ijamec.270374.

[6] J. Memon, M. Sami, R. A. Khan and M. Uddin, "Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR)," in IEEE Access, vol. 8, pp. 142642-142668, 2020, doi: 10.1109/ACCESS.2020.3012542.

[7] Nwakanma, Cosmas & Oluigbo, Ikenna & Izunna, Okpala. (2014). Text – To – Speech Synthesis (TTS). 2. 154-163.

[8] M. H. O'Malley, "Text-to-speech conversion technology," in Computer, vol. 23, no.8, pp. 17-23, Aug. 1990, doi: 10.1109/2.56867.

[9] A. B. M. Ashikur Rahman et al., "Two Decades of Bengali Handwritten Digit Recognition: A Survey," in IEEE Access, vol. 10, pp. 92597-92632, 2022. doi: 10.1109/ACCESS.2022.3202893

[10] Y. He and L. Yang, "Fusion of Human Decision and Artificial Intelligence in Optical Character Recognition through Mobile Phone Interaction," 2022 7th International Conference on Automation, Control and Robotics Engineering (CACRE), Xi'an, China, 2022, pp.

[11] C. Cuenca-Enrique, L. Del-Río-Carazo, E. Acquila-Natale and C. Macarrón, "Semi-automatic document processing and text-recognition for digital transformation of low-resources organizations," 2022 17th Iberian Conference on Information Systems and Technologies (CISTI), Madrid, Spain, 2022, pp. 1-7. doi: 10.23919/CISTI54924.2022.9820135

[12] R. Raj and A. Kos, "A Comprehensive Study of Optical Character Recognition," 2022 29th International Conference on Mixed Design of Integrated Circuits and System (MIXDES), Wrocław, Poland, 2022, pp. 151-154. doi: 10.23919/MIXDES55591.2022.9837974

[13] W. Khallouli, R. Pamie-George, S. Kovacic, A. Sousa-Poza, M. Canan and J. Li, "Leveraging Transfer Learning and GAN Models for OCR from Engineering Documents," 2022 IEEE World AI IoT Congress

(AIIoT), Seattle, WA, USA, 2022, pp. 015-021. doi: 10.1109/AIIoT54504.2022.9817319

[14] S. Ghulyani, D. Jain, P. Singh, S. Joshi and A. Ahlawat, "A Data Entry Optical Character Recognition Tool using Convolutional Neural Networks," 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET), Arad, Romania, 2022, pp. 721-728. doi: 10.1109/GlobConET53749.2022.9872395

[15] A. Yavariabdi, H. Kusetogullari, T. Celik, S. Thummanapally, S. Rijwan and J. Hall, "CArDIS: A Swedish Historical Handwritten Character and Word Dataset," in IEEE Access, vol. 10, pp. 55338-55349, 2022. doi: 10.1109/ACCESS.2022.3175197

[16] S. K. Singh, R. Alam, L. Sujihelen, J. G. L. K, M. P. Selvan and S. Jancy, "Handwritten Character Recognition using Convolutional Neural Network," 2022 6th International Conference on Trends in Electronics and Informatics (ICOEI),Tirunelveli, India, 2022, pp. 1108-1112. doi: 10.1109/ICOEI53556.2022.9777140

[17] N. Teslya and S. Mohammed, "Deep Learning for Handwriting Text Recognition: Existing Approaches and Challenges," 2022 31st Conference of Open Innovations Association (FRUCT), Helsinki, Finland, 2022, pp. 339-346. doi: 10.23919/FRUCT54823.2022.9770912

[18] A. Amanlou, A. A. Suratgar, J. Tavoosi, A. Mohammadzadeh and A. Mosavi, "Single-Image Reflection Removal Using Deep Learning: A Systematic Review," in IEEE Access, vol. 10, pp. 29937-29953, 2022. doi: 10.1109/ACCESS.2022.3156273

[19] V. Rajasekar, J. Premalatha, V. Rangaraaj, B. Rajaraman and A. O. S. Prakash, "Efficient Handwriting Character Recognition Based on Convolutional Neural Network," 2022 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, India, 2022, pp. 1-5. doi: 10.1109/ICCCI54379.2022.9741072

[20] H. Huda, M. Ariful Islam Fahad, M. Islam and A. K. Das, "Bangla Handwritten Character and Digit Recognition Using Deep Convolutional Neural Network on Augmented Dataset and Its Applications," 2022 16th International Conference on Ubiquitous Information Management and Communication (IMCOM), Seoul, Korea, Republic of, 2022, pp. 1-7. doi: 10.1109/IMCOM53663.2022.9721634

[21] S. Van Nguyen, D. A. Nguyen and L. S. Q. Pham, "Digitalization of Administrative Documents A Digital Transformation Step in Practice," 2021 8th NAFOSTED Conference on Information and Computer Science (NICS), Hanoi, Vietnam, 2021, pp. 519-524. doi: 10.1109/NICS54270.2021.9701547

[22] M. M. Islam, A. Das, I. Kowsar, A. K. M. Shahariar Azad Rabby, N. Hasan and F. Rahman, "Towards building a Bangla text recognition solution with a Multi- Headed CNN architecture," 2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, 2021, pp. 1061-1067. doi: 10.1109/BigData52589.2021.9671653

[23] J. D. B. Castro, S. W. A. Canchumuni, C. E. M. Villalobos, F. C. Cordeiro, A. M.
A. Alexandre and M. A. C. Pacheco, "Improvement Optical Character Recognition for Structured Documents using Generative Adversarial Networks," 2021 21st International Conference on Computational Science and Its Applications (ICCSA), Cagliari, Italy, 2021, pp. 285-292. doi: 10.1109/ICCSA54496.2021.00046

[24] A. S. A. Rabby, M. M. Islam, Z. Islam, N. Hasan and F. Rahman, "Towards Building A Robust Large-Scale Bangla Text Recognition Solution Using A Unique Multiple-Domain Character-Based Document Recognition Approach," 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), Pasadena, CA, USA, 2021, pp. 1393-1399. doi: 10.1109/ICMLA52953.2021.00225

[25] X. Zhi, B. Zhao and Y. Wang, "A Hybrid Framework for Text Recognition Used in Commodity Futures Document Verification," 2021 6th International Conference on Computational Intelligence and Applications (ICCIA), Xiamen, China, 2021, pp. 143-148. doi: 10.1109/ICCIA52886.2021.00035