

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/358405430>

A Survey on Scanned Receipts OCR and Information Extraction

Preprint · February 2022

DOI: 10.13140/RG.2.2.24735.84643

CITATIONS

2

READS

5,058

5 authors, including:



Jason Antonio

1 PUBLICATION 2 CITATIONS

SEE PROFILE



Aditya Rachman Putra

Bandung Institute of Technology

5 PUBLICATIONS 19 CITATIONS

SEE PROFILE



Harits Abdurrohman

Bandung Institute of Technology

6 PUBLICATIONS 24 CITATIONS

SEE PROFILE



Moch Shandy Tsalasa Putra

University of Muhammadiyah Malang

2 PUBLICATIONS 14 CITATIONS

SEE PROFILE

A Survey on Scanned Receipts OCR and Information Extraction

Jason Antonio*, Aditya Rachman Putra*, Harits Abdurrohman*, Moch Shandy Tsalasa Putra*, Andreas Chandra*

Jakarta Artificial Intelligence Research, Jakarta, Indonesia

Abstract. Recent advances in deep learning and computer vision as well as natural language processing have been tremendously transformed and reshaped the way machines process unstructured data. Scanned receipts OCR and Information Extraction (SROIE) are the fields that are the intersection between computer vision and natural language processing. SROIE is a field that provides an end-to-end process of recognizing text from scanned images such as receipts and extracting and storing them in a structured format. SROIE contributes a vital role in many document intelligence applications and has high potential in business. The purpose of the systematic literature review is to analyze and summarize current research on the techniques and resources as well as to provide directions for future research. In this paper, we review different techniques for text detection, text extraction, and information extraction for both scanned documents and camera-captured documents written in English and several other languages as well as incorporate methodologies present in the existing research to provide a go-to paper for researchers and practitioners developing this research area.

Keywords: OCR, Scanned Receipt, Information Extraction.

1 Introduction

Scanned receipts OCR is one of the computer vision tasks that specifically extract and recognize text from scanned structured and semi-structured receipts. Many other tasks rely on extracting text to a structured format that can be saved into a database system and cover further tasks such as archiving, image indexing, and financial analytics. Scanned Receipt OCR and Information Extraction (SROIE) is an end-to-end research field that not only extracts information into text but also parses the text into meaningful information. SROIE plays an important role in streamlining document-intensive processes and automation in many financial, accounting, and taxation areas. Recent advances in deep learning bring SROIE in fast and robust development; Many problems that are often encountered are handled using deep learning. There is a leaderboard that focuses on scanned receipt OCR and information extraction that tracks state-of-the-art

* equal contribution

Correspondence email andreas[at]jakartaresearch[dot]com

methods for this task. However, there is a lack of a comprehensive survey that lists, recognizes, and analyzes existing literature and issues that still remain in this field.

There are several excellent review papers in the fields of text detection and recognition as well as optical character recognition. Some of them are outdated, and some are more focused on scene text detection. There are some worth mentioning previous works in the field. [1] presents the first two stages, text localization and extraction in particular, which extract the text before they are fed into an OCR engine. The author listed and summarized natural scene images literature from 2000 up to 2012 which then highlighted state-of-the-art methods as well as common performance measurements. [2] published in 2017, covers up-to-date works before 2017, identifies state-of-the-art deep learning architecture and algorithms as well as potential research directions. In addition, the paper provides comprehensive links to publicly available resources such as datasets, source code, and demonstration. [3] published in 2020, the paper reviews optical character recognition models as well as documents understanding in a short description fashion. [4] provides a comprehensive overview of scene text detection and recognition and highlights the key techniques from handcrafted features and feature learning using convolution. [5] is one of the nearest of our work which was published in 2021, the paper summarizes and analyzes the major changes and significant progress in the field especially using the deep learning approach. [6] attempts to comprehensively review the field of scene text recognition and establish a baseline for a fair comparison of the relevant algorithms. The paper presents the entire picture of scene text recognition by summarizing fundamental problems and state of the art, introducing new insights and ideas, and discussing future trends.

Different from the previous review papers, this study provides a comprehensive survey on optical character recognition on scanned receipts as well as its information extraction. To the best of our knowledge, none of the research examines this specific area of study.

The main contributions of this paper are summarized as follows: 1) summarizes and highlights significant progress in the field of SROIE (2) gives a comprehensive survey on resources such as datasets, source code, and metrics (3) recommends for future research direction and the remain challenges.

The remaining parts of this paper are arranged systematically: In Section 2, we briefly review the preprocessing and data augmentation step for text detection and recognition. In Section 3, we list and summarize the approaches and architectures used for text detection in chronological order. In Section 4, text recognition architectures are listed and analyzed how recent works have been done. In Section 5, we review recent information extraction methods for scanned receipts. Finally, this paper concludes with a discussion of current progress and achievements and future research directions.

2 Data Augmentation and Preprocessing

Data pre-processing and augmentation are two crucial stages in building a robust end-to-end SROIE system. Pre-processing is the first and foremost step for making sure that the data being fed into the OCR system is of sufficient quality for it to distinguish the

characters from the background as easy as possible, hence increasing the character recognition rate. General methods for pre-processing scanned or manually captured images of documents include geometric transformation like rotation and skew correction, binarization, noise removal, and keystone correction. [7] conducted a review on popular skew detection techniques for general document images including Projection Profile analysis, Hough Transform, and Nearest Neighbor. All three techniques are evaluated on the MediaTeam Document Database consisting of Arabic and English Documents. The Projection Profile method is seen to achieve the highest accuracy for skew angle prediction out of the three methods with the other two methods showing very low accuracy. High computation time becomes the trade-off for the accuracy achieved as Projection Profile required the longest execution time with Nearest Neighbor being the fastest out of the three. Both Projection Profile and Hough Transform need relatively extensive computational costs. [8] proposed a novel method that aims to tackle the problems of high computational cost and low accuracy of skew angle estimation. In the aforementioned research, bounding boxes and probability models are used for the documents' slope estimation while Dixon's Q test is combined with the projection profile method to obtain optimal accuracy. The DISEC'13 dataset, consisting of various types of documents ranging from scientific course books to comics, is used and performance is evaluated based on a few criteria including average error deviation and variance of error estimation. This method managed to achieve the lowest average error deviations of 0.23 and fastest runtime of 0.345 seconds compared to Standard Hough Transform (6.12, 2.085 seconds) and Projection Profile method (0.29, 3.776 seconds).

Specific to SROIE-related tasks, as of now, very few research have been found to propose novel pre-processing methods for enhancing the document images for OCR. [9] proposed a deep learning-based method to predict the coordinate of the 4 corners of the document and use the coordinate to perform projective transformation into a rectangular shape which enables the segmentation of the receipt from its background. The skew correction is done in this manner. The architecture of the lightweight deep learning model uses MobileNet as its base model and the whole experiment is performed on a private dataset consisting of various cash receipts images. The method achieves an angular error of 3.22 ± 0.14 . [10] developed a task-based single image super-resolution (SISR) system using U-Net and ResNet 34 model as the base model to increase the resolution of receipt images. The method is implemented on a combination of the ICDAR 2019 dataset, manually selected receipt images, and receipt images from the Scanobar database. It managed to increase recognition up to 15% for poor quality images but degrade well-recognized images by 9%.

Augmentation in OCR systems helps to compensate for the lack of more variety and quantity in the dataset needed for the model to be able to generalize to unseen data. To the best of our knowledge, very little research has been done in augmentation techniques tailor-made for SROIE-related tasks. The scarcity of research on SROIE task-related augmentation methods means that there are ample opportunities for research in this direction. Nonetheless, suggestions for implementation and possibly future more in-depth methods to be researched include geometrically transforming scanned receipt images (affine transformations such as image scaling, rotation, flipping, skewing, brightness, and contrast adjustment,), pixel-level augmentation such as introducing

Gaussian blur, noise, median or mode filters, color jitter, and character-level augmentation including random character replacement, deletion, and insertion.

3 Text Detection

Text detection is one of many steps from SROIE that helps to locate the text from a natural scene, separate text from different entities, detect arbitrary text shapes, and distinct every text or word from a different style. The day before the deep learning era, this area was focused on manually extracting text candidates like CCA (Connected Component Analysis) [11], HOG features with SVM [12], MSER (Maximally Stable Extremal Regions) [13, 14] and even unsupervised learning [15]. This approach leads to a complex pipeline thus deep learning began to be adopted.

Early adoption of deep learning in text detection began with implementing CNN to extract the feature of text from natural scenes. In 2014, Goodfellow et al [16] implemented a multi-digit number recognition from the natural images using DistBelief. This very first approach integrates three separate steps (localization, segmentation, and recognition) using deep convolutional neural networks. This approach worked pretty well on SVHN (Street View House Number) with an accuracy of 96.03%. To the best of our knowledge this approach was the very first to integrate complex pipelines that occur during the pre-deep learning era. Later, several approaches come by separating one or two steps to gain more clarity on the feature map.

We group our collected papers into three distinct categories. 1) text detection only; approach by detecting and localizing text in natural images. 2) end-to-end which consists of text detection and text recognition. An end-to-end method is hard to be separated from text recognition, but we will only cover the detection part. This part mostly consists of a pooling layer, not an entire pipeline. 3) pre- or post-processing, like binarization, weakly supervised learning. This part has the smallest sample, but it leverages some disadvantages from other approaches. The main goal of this section is to improve the quality of the model, either by training strategies or the auxiliary post-processing phase.

3.1 Text detection only

Text detection only is a model that only retrieves the location of the text from a natural scene. The difference between this model and end-to-end is that this approach doesn't have any text recognition module. Based on early adoption, this approach is developed from a fully connected layer, which is quite similar to image segmentation from computer vision tasks.

In 2016 [17] Proposed the idea of using semantic segmentation for text detection. The model produced a pixel-wise prediction map that detects the text. It uses a modified HED that consists of 5 stages in VGG-16 architecture. For each stage, the feature map is projected and fused into a single output. The loss function is produced from this projected feature and ground truth

DMPNet [18] detects and localizes text by using quadrilateral sliding windows in several specific intermediate convolutional layers and filters the proposed areas using the Monte-Carlo method. DMPNet uses several sliding windows that are based on the textual intrinsic shape to fit the ground truth. The overlapping window might occur in the first place and be filtered using a shared Monte-Carlo method.

Image segmentation is highly adopted in text detection too, for example [19] adopts this approach to produce text instance segmentation. The text detection algorithm consists of two stages, a multi-scale FCN for text block extraction and text segmentation. First, Text Line CNN (TL-CNN) produces a segmentation that corresponds to the center of each text line. The output of TL-CNN is split into several images with each image corresponding to a single text line. Then Instance-Aware CNN (IA-CNN) produces a segmentation mask from the text line input. These processes are done in a multi-scale manner. Using three different sizes, the result of each branch with shared convolutional parameters is fed into two pooling layers to merge the features into final results.

A different approach in text feature representation is applied to [20], a text detection model based on the FCN model with NMS and skip connection-like. Built on VGG-16 with 9 extra layers appended after VGG-16 layers. The text-box layer predicts text presence and bounding boxes. These layers are connected to 6 convolutional layers and predict a 72-d vector that consists of text classification score and offset of 12 defaults bounding boxes. NMS is applied in the end.

The authors of [21] claimed that approaches using shared features for text detection and bounding box could result in degraded performance. Due to incompatibility of the two tasks, the authors proposed to perform classification and regression on rotation-invariant features of different characteristics, extracted by two network branches of different designs named Rotation sensitive Regression Detector (RRD). It follows the main architecture of SSD with modified convolution called oriented response convolution. The authors claimed that text coordinates are sensitive to text orientation and with oriented response convolution, the filter is rotated to produce rotation-sensitive features for regression. RRD uses active rotating filters (ARF) [22] to extract those features.

TextSnake [23] main contribution is in the text representation. TextSnake using a series of ordered disks with each center of the circle is corresponding to the center of the line text to represent the text instance. With this method, TextSnake can effectively detect text in various forms like horizontal or curved. Textsnake backbone is VGG-16 and the geometry attributes are predicted in FCN manners.

CRAFT [24] uses synthetic images to learn character-level annotation due to lack of data and predicts the character-level ground truth from real scene images. The CRAFT model has an encoder-decoder architecture. The backbone is VGG-16 with batch normalization and has a skip connection that is similar to U-net in the decoder. The model is trained in a weakly-supervised manner.

TextFuseNet [25] approaches text detection in more radical ways, combining each feature from character-level, word-level, and global-level to produce novel text representation. The ResNet-FPN as backbone produces features to feed semantic segmentation and RPN branch. The semantic segmentation branch yields a feature for the RPN branch. The RPN branch also known as the detection pipeline splits into two branches, detection branch and mask branch. Character- and word-level feature extraction within

detection and mask branches in the Mask R-CNN pipeline. RoIAlign is applied to extract different features and perform detection for both words and characters. Feature maps of the segmentation branch become global-level features, while feature maps from RPN with RoIAlign produce word-level and character-level features. For each word-level instance, authors fused the corresponding character-, word-, and global-level features within the multipath fusion architecture for instance segmentation. In the end, all features are fused.

Text Localization Generative Adversarial Network (TLGAN) [26] is a model that relies heavily on GAN to produce text localization maps. In the context of SROIE, the location of the text instances is a crucial step to be solved. TLGAN uses pre-trained VGG for both generator and discriminator. The generator has a similar configuration to the semantic segmentation model for text localization like [24]. The generator produces the mask or text localization map, and the discriminator would classify the output with the ground-truth. The text localization is in the post-processing where the true prediction of bounding boxes is classified within the threshold.

3.2 End-to-end Architecture

The end-to-end model consists of text detection and text recognition. Although the main purpose of this approach is to recognize the text, some of the major contributions lie in text detection too. In this section we will only focus on the text detection part which is mostly in the intermediate process of the full pipelines

Connectionist Text Proposal Network (CTPN) [27] is an architecture that directly localizes text sequences in convolutional layers. This method contributes to detect text in fine-scale proposals (finding the text line and multi-scaled features), adds recurrent connectionist text proposals (because the authors treat text as a sequence), and gives side-refinement to accurately estimate the offset for each anchor/proposal in both left & right horizontal sides

The part of text detection from this approach lies in TPN (text proposal network), ROI pooling, and MLP [28]. The proposed model produced text bounding boxes and text labels. The TPN is inspired by RPN to retain both local and contextual information. TPN produced features for classification and regression, to classify whether the proposed RoIs are text or not and refine the coordinates of the bounding box. This refinement is also done in a similar manner at Text Detection Network (TDN). Region feature encoder (RFE) converts the feature from TPN to feed TDN. Then TDN classifies the text and predicts bounding box offsets. Results from TDN are fed to RFE again to produce recognized words from the Text Recognition Network (TRN).

TextSpotter is another end-to-end text detection model using RoIAlign for text alignment and character attention for recognition. The authors [29] proposed text alignment to extract a sequence of features from the quadrilateral region from multi-orientation features. The RoIAlign also provides word-level alignment. The backbone is fully convolutional built on PVA network due to low computational cost and the text detection part consists of two branches on the top convolution layer with two different functions: joint text/non-text classification (returns softmax-ed classification map) and multi-orientation bounding boxes regression (returns five localization maps with the same

spatial size, which estimate five parameters consist of top, bottom, left, right and inclined orientation for each bounding box with an arbitrary orientation at each spatial location of text regions).

FOTS or Fast Oriented Text Spotting is a model that shares features from text detection to text recognition branch [30]. The author combines detection and recognition as a single operation to detect & recognize the text. This approach proven performs better in detection because text recognition supervision helps the network to learn detailed character level features. The pipeline has an encoder-decoder architecture with ResNet-50 as the backbone. The shared convolution is then fed to the text detection branch. This branch yields predicted COCO bounding boxes fashion. The architecture adopts a fully convolutional network with modified heads to reproduce 5 distinct values: top, bottom, left, right, and the orientation of the related bounding box. Final detection results are produced by applying thresholding and NMS. Then the result is fed to RoIRotate to produce a feature map for the text recognition branch. RoI rotation is responsible for creating invariant features.

PixelLink [31] tackles a problem where the text is closed to each other in instance segmentation. Text instances are first segmented out by linking pixels within the same instance together. Text bounding boxes are then extracted directly from the segmentation result without location regression. The backbone, VGG-16 is trained to perform two kinds of tasks: text instance classification and link classification. The text instance classification part predicts whether a pixel within the text instance is labeled as positive (text) or negative (non-text). Link classification predicts a given pixel and its neighbors are in the same instance or not. This part was inspired by SegLink [31]. Both tasks are done in a pixel-wise manner.

An improvement of TextSnake, TextDragon [32] replaces the circular sequences text instance with TCL and local box imbued with orientation. This end-to-end arbitrary text recognition using RoISlide. This method adopts the human eye reading mechanism: find the local area of the text, get the content within the area then eyes move along the centerline of the text. This is done by one area at a time to overcome the variations of character size and orientation. Same as TextSnake, the backbone is VGG-16 and adopts Connectionist Temporal Classification (CTC) [33] for text recognition.

Region Proposal Network (RPN) is one of the most well-known methods that is being used for arbitrary text detection, but the downside of RPN is that this model relies heavily on handcrafted anchors, and its proposal region is presented in a bounding box. SPN or Segmentation Proposal Network is another option for this task. Mask TextSpotter v3 [34] uses SPN as an anchor-free backbone that gives a better text instance representation in a segmentation manner. The anchor-free SPN overcomes the limitations of RPN in handling text of extreme aspect ratios or irregular shapes and provides more accurate proposals to improve recognition robustness. Hard RoI masking is also present in this work to add polygonal to RoI features. The authors claimed to significantly improve robustness to rotations, aspect ratios, and shapes. The text detection part lies on SPN and hard RoI masking.

3.3 Auxiliary Process

The auxiliary process is another part that contributes to the text detection in the manner of creating a richer feature or better outputs. Several approaches lie in this area include weakly supervised learning, semi-supervised learning, binarization, and synthetic datasets utilization.

Differentiable binarization (DB) [[35] performs a post-processing procedure to convert probability maps produced by a segmentation method into bounding boxes/regions of text. This approach is well paired with a lightweight model.

Due to lack of data, several methods like TextDragon, CRAFT, and TextFuseNet can be trained with weakly-supervised learning. This type of training produces pseudo-ground truth either from the interim model or overall model to exploit richer features from ground truth. TextFuseNet uses weak supervision for word-level annotation to guide searching character training samples while TextDragon can be trained in a similar way to improve the model's practicability.

Lastly, synthetic data is used to provide a better-generated dataset for training. Mostly used in pre-training before using a real dataset. SynthText [36] overlays synthetic images into scene images in a natural way. The authors provide a better text rendering, so the text placed perfectly fits the scene. Many methods use this approach in their pre-trained phases like TextDragon, CRAFT, multi-scale FCN, TextSpotter, and FOTS.

4 Text Recognition

Text recognition is the next process in a pipeline after text detection, i.e. after a region is identified that contains a text, the next logical step is to recognize what exactly is the text within.

In past years, there has been rapid progress in the research of text recognition, especially in the domain of handwritten text recognition in multiple characters such as Japanese [37], Chinese [38], and Latin [39] which have their own share of challenges. Since this review focuses on SROIE, which contains printed text, we will generally look into text recognition on receipts or printed documents. In this case, we also look at scene text recognition since most of the texts are printed.

4.1 Recognition Only

One of the approaches without the use of a neural network-based model in text recognition was presented by [40], using a template matching algorithm to match segmented characters of scanned receipts with templates generated from the average of the training images. The result is evaluated on a self-made dataset of scanned receipts from only one retailer on their character recognition rate and achieves a 97.36% recognition rate for all characters (including blanks). Unfortunately, the choice of dataset makes it hard to compare with other methods. Another approach is used in Tesseract OCR (2007) [41] which used polygon approximation of each character as features, then used a look-

up table and distance calculation to classify each character. This approach is still widely used at the time of this writing.

Recently, most approaches were done based on Deep Neural Network (DNN). Starting with a model based on Convolutional Neural Network (CNN) only. One such approach is proposed by [40], who proposed a combination of CNN and with average pooling to perform both text detection and recognition on an image. It performed better compared to the previous approach by [42] on end-to-end evaluation using both ICDAR and SVT datasets.

Another class of approach incorporates encoder-decoder architecture to split the process into two parts. Which consists of an encoder that takes an image of words or text lines and uses a Convolutional Neural Network (CNN) based model to encode the image into a feature map, and sometimes paired with a Recurrent Neural Network (RNN) based model that helps encode the sequential nature of the character in the image. The decoder part uses the encoded image to generate the probable output. Several notable approaches work on modifying the component of this encoder-decoder architecture to make it work better on certain tasks.

Shi et al. [43] first introduced this approach which used CNN to extract feature maps from an image and then pass this feature sequence into a bidirectional LSTM, this model was named as Convolutional Recurrent Neural Network (CRNN). To transcribe the output of CRNN into a predicted sequence, a Connectionist Temporal Classification (CTC) layer is used based on Graves et. al. (2006) [33]. While this paper didn't explicitly mention encoder-decoder architecture, but the overarching idea was still used in more recent papers.

Wang et al. [44] took inspiration from GRU, having a controlling gate that can weaken or cut off irrelevant context information can help models to learn better. Especially mimicking the eye's Receptive Field. RCNN (CNN with Recurrent Convolution Layer) is widely used and has shown impressive results, Gated RCL which surprisingly is the general form of RCL can improve the expressive capability of the model and in turn enable the model to learn better. The result itself doesn't really discuss what can be improved based on the problem faced currently by this model.

NRTR by Sheng et al. [45] analyzed the shortcoming of RNN and CNN-based text recognition model and proposed a model based only on stacked self-attention. While RNN based models could learn contextual information and correlation between characters in the text, the non-parallelizable nature of RNN imposes time and computation burdens when the image is long. CNN circumvents this problem with its receptive field that could be computed at parallel, but it introduces a problem when the model needs to learn the relation between distant positions (unless more convolution layer is added). This paper shows that stacked self-attention as encoder and decoder produced the competitive result on the common benchmark. They argue that it also solved problems found in CNN and RNN based model.

[46] This paper studies the effectiveness of the encoder-decoder network which mainly contains a self-attention network namely Self-Attention Text Recognition Network (SATRN). SATRN uses a self-attention mechanism to capture spatial dependencies of characters in a scene text image.

One of the latest approaches in attention-based models is TrOCR, proposed by Li et. al. [47] seeks to leverage a pre-trained image transformer model for image understanding and text transformer models for workpiece-level text generation. As the name suggests, TrOCR does not use CNN and RNN, it uses multi-head attention for both the encoder and decoder. It also does not use CTC as a transcription layer to produce output tokens from the decoder result. In the [47], multiple pre-trained image transformer models are used, which show that BEiT Large [48] model with RoBERTa Large [49] perform the best compared to other.

Another notable approach is proposed by Wan et. al. [50], namely TextScanner. A model that used RNN-attention-based method to predict the character segmentation of an image. The focus of this approach is the decoding part of the model, which uses two branches to help in transcription. The class branch is used to predict the character segmentation of an image, and the geometry branch is used to predict the order of the character and its position producing an order map. A serious caveat on character segmentation is the needs to have character level annotation which is not as abundant compared to text annotation. To solve this, TextScanner uses mutual supervision (similar to self-supervised learning) which utilize the redundancy introduced by order segmentation, localization map, and character segmentation to simulate character level annotation.

Yu et al. [51] brought into focus the concept of semantics for text recognition in their Semantic Recognition Network (SRN) framework. They argued that most approaches use RNN-like structures that model semantic information albeit implicitly. The framework itself is end-to-end trainable and facilitates parallel training which is not possible when using RNN based methods. They also proposed Parallel Visual Attention Module (PVAM) that pays attention to each character in an image in each attention map (visual features). The output of PVAM will then be converted into semantic embedding and processed to get its semantic features in the Global Semantic Reasoning Module. While the model proposed in this paper are rooted in attention-based model, the explicit separation of visual features and semantic features is the main point of this framework.

4.2 Pre and Post Processing

A variety of papers shows that improving processes around recognition helped in improving recognition performance including a class of papers about preprocessing (rectify) and also post-processing (utilizing language model to improve performance, or domain adaptation [52]). [53] claim that scene text detection and recognition rely on text detection and individual character recognition for distorted text. And faced various challenges because of this problem. To solve this, a robust and accurate distortion rectification is designed by utilizing iterative rectification. Using spline with a vertical line, fitted to the text in the picture and iteratively transform the original image by normalizing the spline, in order to get a normalized text. This method can help in recognizing distorted text (both caused by curvature and perspective), and with an additional dictionary, it can further improve the recognition accuracy. In retrospect to the recognition model, this preprocessing poses a relatively low-performance penalty, comparable to the previous method by Shi et al. [54] without iteration.

[55] Focuses on receipt images (especially groceries). But rather than text recognition, the paper itself discusses preprocessing, postprocessing, and Information Extraction. Their improvement stems from several rules that help the output of commercial OCR Engines to discern information from grocery receipts. Using generic OCR, together with robust preprocessing and several rule-based algorithms for short-form words conversion, unwanted text removal, garbage text removal (based on heuristics), information extraction using regex, and item name correction using grocery dictionary by fuzzy search can improve the result of this specialized task.

Another approach focusing on preprocessing was proposed by Shi (2015) [43]. This paper discusses preprocessing step for scene text recognition in which the text in the image is having perspective and curvature distortions. This paper proposes an Iterative Rectification Network that is able to transform from curved text into a horizontal text, and the transformed image fed into a text recognition network which consists of encoder-decoder architecture. The recognition network benchmarks 2 common backbones for decoder namely ResNet and VGG followed by BiLSTM and the Decoder uses LuongAttention mechanism which consists of 2-layer attentional LSTM. The paper also uses beam search during inference in the decoder stage.

NRTR also introduced preprocessing step to effectively convert a 2D image into a 1D sequence, which is usually performed by patches like in image transformer models (BEiT), instead, a stacked convolutional layer is used to get a 1D sequence from 2D images. This paper shows that more convolutional layer didn't improve the performance, as 2 convolutional layer shows the best result for a large model. While the smaller model may benefit from the CNNLSTM model inspired by speech recognition [56].

Graph Convolutional Network for Textual Reasoning (GTR), proposed by He et. al. [57] is a module that can be plugged into STR to improve the performance by providing textual reasoning. Rather than an end-to-end model, GTR can be fed a character segmentation maps which can be produced by a visual recognition (VR) model. And uses that information to construct a graph which denotes the relation between each segment. This paper also introduced S-GTR, an implementation of GTR on VR based on RNN and Language Model based on SRN.

4.3 End-to-End

End-to-End approaches that process images with multiple images and only output information which rather than an end-to-end recognition, it's actually an Information Extraction method. [58] proposes EATEN, an end-to-end visual text extraction based on LSTM. Rather than using text detection and recognition, the task itself is focused on extracting a set of defined Entity of Interest from documents of a specific domain (in this case identity card) such as Name, Social Security Number, Date of Birth, etc. This model can be trained without direct annotation inside the image, instead, a label for the whole image is sufficient to extract the EoI. This kind of approach can utilize a huge synthetic data for training given the document type and EoI is clearly defined, but it means the approach is difficult to generalize and to be explained.

He et. al. [29] proposed a framework to train both text detection and recognition in a unified framework. It is done by incorporating text alignment after detection to rectify the image and character attention mechanism is applied in decoding process to help the recognition of each character. While this paper discussed both detection and recognition, we'll focus on the recognition part which received a sample grid from text detection steps of the architecture. This convolutional feature of the sample grid will be fed into an inception network and bidirectional LSTM layer in the encoding part. Then a character attention mechanism is used on the decoding part to guide the process to focus on each character and prevent misalignment of the text.

5 Information Extraction

The next stage in the SROIE pipeline is to parse and extract meaningful information from the unstructured recognized text found in receipts into a structured format that is more explainable. This includes finding and explaining relationships between entities in the text data, so precise information can be obtained without having to go through all the extracted text manually. General documents may have many different layouts and formats, including variable font size and family, table arrangements, multiple columns, etc. The layouts, positioning, and sizing of texts are necessary to obtain comprehensive semantic content and information from the scanned documents. In addition to the diverse layout and formats, the poor quality of the scanned images and the complexity of document template structures make understanding documents a very challenging task. The more conventional approach, which may involve using regex, as shown in [59], and template matching combined with other techniques as shown in [60] and [61] have been used before. However, increasingly popular Deep learning-based methods have now been used to tackle the complexity of document structure removing the need for determining the features to be extracted manually.

In visually rich documents (VRDs), both visual and layout information is important for understanding the contents. To extract key information in visually rich documents requires understanding the contextual and semantics of text present in the document images in 2D space. Exploring more visual and textual features of the documents becomes the main challenge in order to get richer semantic representation without ambiguity.

Several Information Extraction methods have been proposed, which will be discussed in this paper, even though many of them are performed for general document analysis (e.g., Form, Invoice). Methods that have been included for discussion, generally use the encoder-decoder model and BERT-language model as part of its base architecture. Various parsing methods to ensure that spatial and visual features which may include layout structure along with 2-D positions of text in the scanned receipts relative to each other) and textual features to be preserved. Based on the backbone model/architecture used, each method/proposed solution will be discussed.

Table 1. Summary of the surveyed performance on ICDAR 2019 SROIE dataset

Model	F1-Score	Precision	Recall	Accuracy
LayoutLM	0.95	0.95	0.95	0.95
LayoutLMv2	0.97	-	-	-
BROS	0.96	-	-	-
PICK	0.96	-	-	-
TCPN-T	0.95	-	-	-
TRIE	0.96	-	-	-

Table 2. Summary of the surveyed performance on CORD dataset

Model	F1-Score	Precision	Recall	Accuracy
DeepCPFG	0.92	0.95	0.95	0.95
LayoutLMv2	0.96	-	-	-
SPADE	0.91	-	-	-
BROS	0.97	-	-	-

5.1 Chargrid

Katti et al. [62] proposed a novel paradigm for processing and understanding structured documents called Chargrid. Instead of converting a document into a 1D text, Chargrid preserves the spatial structure of the document by representing it as a sparse 2D grid of characters. The shape of Chargrid is a 2D layout and constructed from character boxes, like bounding boxes surrounding each character from document pages. Its architecture contains an encoder-decoder network. The encoder network is similar to a VGG-type network with dilated convolutions and the decoder network will be split into two branches where each will be used for Semantic Segmentation and Bounding box regression respectively. In this research, Katti et al. used a Private 12K Invoice dataset and use accuracy as a metrics evaluation. Pure chargrid-net has the highest accuracy value when compared to the other model. Katti et al. expect hybrid models will have the highest accuracy value but turns out it doesn't add any benefits.

Timo and Christian [63] proposed combining Chargrid architecture with BERT language model to construct a grid on the word-piece level and embed it with dense contextualized vectors. Timo and Christian use the same semantic segmentation and bounding box regression as in Chargrid but use BERTgrid tensor as the input to the neural network. Timo and Christian use the same dataset and metrics evaluation as in Katti et al. BERTgrid when compared to Chargrid has significant improvements from $61.76\% \pm 0.72$ to $65.48\% \pm 0.58$

5.2 LayoutLM

Xu et al. [64] proposed a pretraining method of text and layout for document image understanding tasks like information extraction by using BERT as the backbone model. Even though BERT models can tackle several challenging NLP tasks, when it comes to visually rich documents, there is much more information that can be fed into BERT pretrained models, like Document Layout Information and Visual Information. Therefore, Xu et al. use BERT architecture as the backbone and add two types of input embeddings, 2-D Position embedding for modeling spatial position in documents and in bounding box shape and Image embedding is for representing image features in language shape. The reason why Xu et al. add two input embeddings is because 2-D position embedding can capture relationships between tokens and a document, while image embedding can capture other features from document images like font directions, types, and colors. Tasks that are conducted by Xu et al. in this research are divided into two Pre-training and Fine-tuning. For the Pre-training task, IIT-CDIP Test Collection 1.0 dataset is used, and for Fine-tuning tasks, FUNSD, SROIE, and RVL-CDIP datasets are used. Metrics evaluation used in this research are Precision, Recall and F1 Score for FUNSD and SROIE datasets and Accuracy for RVL-CDIP dataset. LayoutLM managed to achieve 0.7677 in precision, 0.8195 in recall, 0.7927 in F1 Score for FUNSD dataset. Performance on SROIE dataset showed 0.9524 in precision, 0.9524 in recall, and F1 Score of 0.9524. For RVL-CDIP dataset, LayoutLM achieved an accuracy of 94.42%. Experiments done by Xu et al. show that LayoutLM can outperform several SOTA pre-trained models in these three tasks.

Sage et al. [65] proposed that LayoutLM can do high sample efficiency when fine-tuned on public and real-world Information Extraction datasets, and for that reason Sage et al. compare 3 models that are consisting of an Encoder to create contextualized representations of the tokens and Decoder that decodes sequence of representations into extract information, all models only differ in their Encoder part. First model is Pre-trained LayoutLM, second model is Fully supervised models that have 2 encoders, reuse LayoutLM model but without using pre-training and randomly initialize all parameters, secondly using 2-layer BiLSTM network, and last model will be built from scratch. Sage et al. using two Information Extraction datasets that cover different document types and extraction objectives, SROIE and PO-51k. The results are for SROIE Pre-trained model F1 Score 0.9417 and BiLSTM F1 Score 0.8874. Sage et al. also note that fine tuning on SROIE can improve F1 Score up to 10% from PO-51k dataset for 8 documents, and fine tuning before SROIE dataset can reduce the variance.

Chua and Duffy [66] proposed the DeepCPCFG method that can define a grammar for Information Extraction without full description from document layout. By using Context Free Grammar (CFG) to extract information that allows to capture even more complex information from document structures. Information is extracted by parsing documents to CFG in 2D regions. Architecture model proposed by Chua and Duffy is using LayoutLM for Encoder that receives input bounding box coordinates and text that later will produce word embedding for every bounding box, and for Decoder will produce parse trees that reflect the document hierarchy. In this research will be using CORD and RVL-CDIP datasets for conduct the experiments, the result for CORD

dataset is measured with F1 Score using SPADE (Spatial Dependency Parsing) metrics 92.2.

Xu et al. [67] proposed an improved version from the previous work by using another architecture with new pretraining tasks to model interaction with text, layout, and image in a single multi-modal framework. Another difference between vanilla LayoutLM and LayoutLMv2 is visual embeddings combined in the fine-tuning stage by integrating visual information in the pre-training stage with Transformer architecture to learn the cross-modality interaction between visual and textual information as input. Tasks that are conducted in this research are the same as that of [LayoutLM paper], divided into Pre-training and Fine-tuning stage as before. Same datasets are used but in the Fine-tuning task, there are additional datasets like CORD, Kleister-NDA, and DocVQA. Overall, the results show that LayoutLMv2 outperforms LayoutLM by a large margin and achieves new SOTA results on a variety of visually rich document understanding tasks.

5.3 Graph-based Method

Hwang et al. [68] proposed a novel information extraction method called SPADE that can handle complex spatial relationships or spatial layout and hierarchical information structure from semi-structured document images. To create a better model for spatial relationship and hierarchical information from semi-structured document images, Hwang et al. create a spatial dependency parsing task by constructing a dependency graph with tokens and fields as the graph nodes (node per token and field type). SPADE architecture model are consist from (1) Spatial Text Encoder based on 2D Transformer architecture, (2) Graph Generator, how it is work is every token corresponds to a node and each pair of the nodes forms one of the two relations (or no relation), and (3) Graph Decoder is a deterministic function to maps the graph to a valid parse of output structure. In this research, Hwang et al. using two types of datasets, Public and Private datasets, for public are using CORD and FUNSD datasets, and for private are using RECEIPT-IDN, NAMECARD (JPN), and INVOICE(JPN) datasets. SPADE achieves 91.5% and 87.4% in F1 Score with and without the oracle (ground truth OCR results).

Hong et al. [69] proposed a pre-trained language model that combined BERT and combinations of texts and spatial information without relying on text blocks or visual features. Hong et al. adopted SPADE decoder to create relation graphs of tokens to represent key entities and relationships in Key Information Extraction (KIE) tasks. The main architecture of BROS follows LayoutLM, but there are two major updates: (1) using of spatial encoding metric that can describe spatial relations between text blocks and (2) using 2D pre-training objective for text blocks 2D space. Hong et al. using FUNSD, SROIE, CORD, and SciTSR to do performance comparisons for Entity Extraction (EE) and Entity Linking (EL) task with “with” and “without” the order of information of text blocks scenario.

Liu et al. [70] proposed a Graph Convolution based model by combining textual and visual information from Visual Rich Document (VRD). Graph embedding is produced by Graph Convolution that later will perform summarizing context from text segment document, later Graph embedding will be combined with standard BiLSTM-CRF

model. The embedding represents the information for visual and textual context, that means for visual context refer to the layout of the document and relative position of the individual segment to other segments and for textual context is the aggregate of text information in the document. Graph convolution is applied to compute visual text embeddings of text segments. There are some components from VRD that are useful, like color and font family, because it is complementary to the visual and textual context. Liu et al. are using Value Added Tax Invoices (VATI) and International Purchase Receipts (IPR) datasets, and the results are F1 Score for VATi 0.873 and IPR dataset 0.836.

Yu et al. [71] proposed a PICK method using Graph Convolution Network that will be using all the features from document, text, image, and position features to get more richer representation for Key Information Extraction (KIE) and to improve extraction ability. PICK combining Graph module with Encoder-Decoder framework, for Encoder using Transformers for text embedding and Convolutional Neural Network for image embedding. Yu et al. using two types of datasets, Private with Medical Invoice and Train Ticket datasets and Public with SROIE dataset. PICK achieved the result with Medical Invoice with average F1 Score 87.0, Train Ticket 98.6, and SROIE 96.1. PICK can handle extraction tasks very well on fixed layout documents due to PICK having the ability to learn the graph structure of documents.

5.4 End-to-end

Dang et al. [72] proposed a Multi-Stage Attentional U-Net (MSAU) method that can perform end-to-end information extraction with pixel-level semantic segmentation tasks with Chargrid to label and extract the relevant information. To be able to correctly exploit textual and spatial features from Chargrid, MSAU apply multi-stage encoder-decoder design. Dang et al. use the backbone Coupled U-Net for MSAU architecture, as it can prevent gradient vanishing. This paper also used data augmentation processes, like random character replacement to mimic OCR errors, random text shifting, random affine transformation, random background padding to enhance the generalization performance of MSAU model. Dang et al. are using self-collected Japanese invoices and the medical receipts dataset for this research, metrics evaluation that used are mean Intersection-over-Union (mIOU), mean pixel accuracy (pix acc) and box F1-score (F1-Score). Overall, the MSAU big model got the best result when compared to the base model with mIOU 87.2, pix acc 92.5 and F1-Score 96.0 for Japanese invoices, and mIOU 89.1, pix acc 93.3 and F1-Score 96.1


Wang et al. [[73] proposed an end-to-end weakly-supervised learning framework for visual information extraction task called TPCN (Tag, Copy or Predict Network). It made use of a flexible decoder that can combine weakly-supervised training strategy and have two switchable modes, Tag Mode (TCPN-T) and Copy or Predict Mode (TCPN-CP) for missing or wrong tokens, this mode is more suitable for sequences of categories with strong semantic relevance. For end-to-end scenarios also have Text Detector and Text Recognizer. Authors also proposed a TextLattice method that can reorganize OCR results to 2D document representation. Architecture in TPCN framework for Encoder is using ResNet + U-Net & BiLSTM in attention mechanism for Decoder.

Wang et al. are using SROIE and EPHOIE datasets with F1-Score as metrics evaluation, for SROIE got 96.54 and EPHOIE got 98.06.

Zhang et al. [74] proposed an end-to-end trainable information extraction framework that can handle various types of VRDsm from structured to semi-structured documents by bridging Text Reading and Information Extraction tasks with shared feature, including position features, visual features, and textual features with Multimodal Context Block, those two stages are separately executed. The Text Reading module is responsible for localizing and recognizing all texts in document images, and Information Extraction module is to extract entities of interest from document images. Framework in this research is adopting ResNet and Feature Pyramid Network (FPN) as their backbone. Zhang et al. are using SROIE public dataset and Taxi & Resume private datasets. Metrics evaluation that used is F1-Score, for SROIE have 2 settings, setting 1 is for prediction of boxes and transcript of texts and setting 2 is for ground-truth of boxes and transcript of texts. F1-Score for Setting 1 82.06 and Setting 2 96.18.

6 Dataset

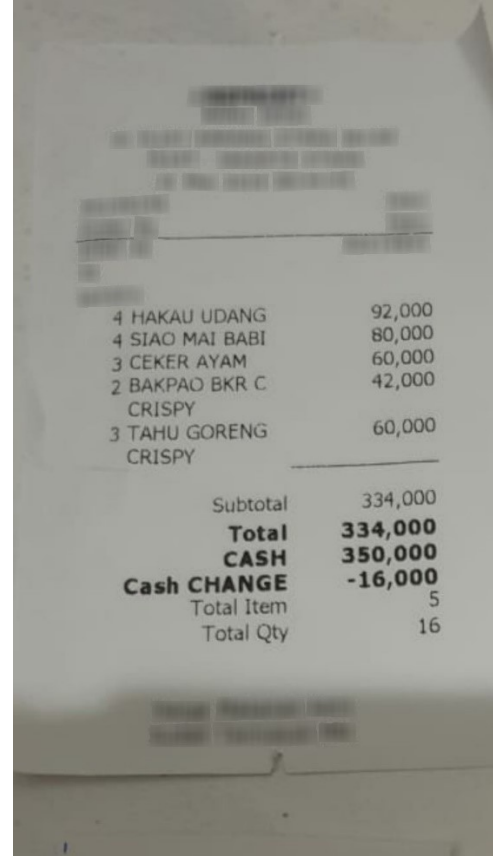
Table 3. ICDAR 2019 Scanned Receipt OCR and Information Extraction Dataset. Left is a sample image and right is the JSON format. The challenge has 3 tasks. Task 1 and Task 2 are for text detection and recognition. Task 3 is information extraction.

	<p>Task 1 & 2:</p> <ul style="list-style-type: none"> - 72,25,326,25,326,64,72,64,TAN WOON YANN - 50,82,440,82,440,121,50,121,BOOK TA .K(TAMAN DAYA) SDN BND - 165,372,342,372,342,389,165,389,25/12/2018 8:13:39 PM - 110,144,383,144,383,163,110,163,NO.53 55,57 & 59, JALAN SAGU 18 - 412,639,442,639,442,654,412,654,9.00 <p>Task 3:</p> <pre>{ "company": "BOOK TA .K (TAMAN DAYA) SDN BHD", "date": "25/12/2018", "address": "NO.53 55,57 & 59, JALAN SAGU 18, TAMAN DAYA, 81100 JOHOR BAHRU, JOHOR.", "total": "9.00" }</pre>
--	--

[75] SROIE ICDAR 2019 is one of the most used dataset for receipt OCR task. ICDAR conducted a competition that consisted of three tasks and each task have its own leaderboards and metrics. The three tasks are text localization, text recognition, and information extraction. The dataset contains 1000 images which are then split into train and

test for 600 and 400 respectively. The test set is used for submission and as a benchmark in the leaderboard. The dataset for task 1 and task 2 contains coordinates of the bounding box and the text while task 3 contains the text and its key information.

Table 4. A sample image taken from CORD dataset which contains more fine-grained information extraction.

	<pre> { "words": [{ "quad": { "x2": 272, "y3": 489, "x3": 270, "y4": 489, "x1": 174, "y1": 461, "x4": 174, "y2": 459 }, "is_key": 0, "row_id": 539268, "text": "HAKAU" }, { "quad": { "x2": 379, "y3": 488, "x3": 380, "y4": 488, "x1": 280, "y1": 460, "x4": 278, "y2": 457 }, "is_key": 0, "row_id": 539268, "text": "UDANG" }], "category": "menu.nm", "group_id": 3 } </pre>
--	--

[76] CORD Receipt dataset contains 11,000 Indonesian receipt images that were obtained from shops and restaurants. Each image has its own corresponding JSON which contains the coordinates of bounding boxes, text, and fine-grained category. In general, CORD has 9 super classes and 54 subclasses. The super classes tag main information in the receipt such as store info, menu, total, etc. while subclasses are the detailed part of the superclass such as menu name, quantity, price, etc.

7 Conclusion

As stated previously, Scanned receipts OCR and Key Information Extraction (SROIE) harnesses immense potential in terms of benefitting the overall work pipeline in various areas like finance, accounting, taxation, and other business-related archiving through automation. In this paper, the three key-tasks in SROIE: Text Detection, Text Recognition, Information Extraction, are discussed and several state-of-the-art methods are reviewed. Even though most of the methods discussed in Text Detection and Text Recognition section were developed for Scene Text Detection and Recognition tasks, the challenges that these methods were trying to solve can be extrapolated to the domain of SROIE. Challenges related to SROIE includes blurred or low-resolution scanned receipt, faded texts, varying structure and complexity of layouts, relatively long text compared to those in STR tasks, rolled and creased papers, very small font sizes, poor printing quality, unneeded texts in background in addition to much larger accuracy required for SROIE to be beneficial. We look into some notable methods reviewed in earlier sections that can be implemented to solve challenges in SROIE and assess the open issues that remains exists.

In text detection and recognition, we recognize 3 things that are fundamental to the development of the architecture. 1) **Text line Length Limitation**. Most STR research uses relatively small text line character length, while some offered framework on how to extend the method into longer text line like TextScanner and SRN, but it also demanded significant additional computation. 2) **Warped Text**. Since longer receipts have tendencies to curve, models with rectification step may offer better performance, especially end-to-end methods like TextSpotter which has rectification built in, or ESIR which provides iterative rectification of text line image. 3) **Rise of Semantics**. Semantic Recognition Network brought semantics into text recognition, especially visual semantics, with promising result. While several papers down the line also use textual semantics like SRN GTR, but its reliance in LM begs the question of its usefulness in the setting of SROIE (with domain specific lexicon).

References

1. Zhang H, Zhao K, Song YZ, Guo J (2013) Text extraction from natural scene image: A survey. *Neurocomputing* 122:310–323. <https://doi.org/10.1016/j.neucom.2013.05.037>
2. ZHU Y, YAO C, BAI X (2017) Scene text detection and recognition: recent advances and future trends. *Frontiers of Computer Science* 10:19–36
3. Subramani N, Matton A, Greaves M, Lam A (2020) A Survey of Deep Learning Approaches for OCR and Document Understanding. In: 34th NeurIPS 2020 Workshop: ML Retrospectives, Surveys & Meta-analyses (ML-RSA).
4. Lin H, Yang P, Zhang F (2020) Review of Scene Text Detection and Recognition. *Archives of Computational Methods in Engineering* 27:433–454. <https://doi.org/10.1007/s11831-019-09315-1>

5. Long S, He X, Yao C (2021) Scene Text Detection and Recognition: The Deep Learning Era. *International Journal of Computer Vision* 129:161–184. <https://doi.org/10.1007/s11263-020-01369-0>
6. Chen X, Jin L, Zhu Y, et al (2021) Text Recognition in the Wild: A Survey. *ACM Computing Surveys* 54:. <https://doi.org/10.1145/3440756>
7. Al-Khatatneh A, Pitchay SA, Al-Qudah M (2016) A Review of Skew Detection Techniques for Document. *Proceedings - UKSim-AMSS 17th International Conference on Computer Modelling and Simulation, UKSim 2015* 316–321. <https://doi.org/10.1109/UKSim.2015.73>
8. Huang K, Chen Z, Yu M, et al (2020) An efficient document skew detection method using probability model and Q test. *Electronics (Switzerland)* 9:. <https://doi.org/10.3390/electronics9010055>
9. Dobai L, Teletin M (2019) A document detection technique using convolutional neural networks for optical character recognition systems. *ESANN 2019 - Proceedings, 27th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning* 547–552
10. Robert V, Talbot H (2020) Does Super-Resolution Improve OCR Performance In The Real World? A Case Study On Images Of Receipts. In: *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, pp 548–552
11. Wang X, Song Y, Zhang Y (2013) Natural scene text detection with multi-channel connected component segmentation. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR* 1375–1379. <https://doi.org/10.1109/ICDAR.2013.278>
12. Fabrizio J, Marcotegui B, Cord M (2013) Text detection in street level images. *Pattern Analysis and Applications* 16:519–533. <https://doi.org/10.1007/s10044-013-0329-7>
13. Li Y, Lu H (2012) Scene text detection via stroke width. *Proceedings - International Conference on Pattern Recognition* 681–684
14. Yin XC, Yin X, Huang K, Hao HW (2014) Robust text detection in natural scene images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36:970–983. <https://doi.org/10.1109/TPAMI.2013.182>
15. Coates A, Carpenter B, Case C, et al (2011) Text detection and character recognition in scene images with unsupervised feature learning. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR* 440–445. <https://doi.org/10.1109/ICDAR.2011.95>
16. Goodfellow IJ, Bulatov Y, Ibarz J, et al (2014) Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks. In: *International Conference on Learning Representations*. pp 1–12
17. Yao C, Bai X, Sang N, et al (2016) Scene Text Detection via Holistic, Multi-Channel Prediction. In: *arXiv:1606.09002v2 [cs.CV]*
18. Liu Y, Jin L (2017) Deep matching prior network: Toward tighter multi-oriented text detection. In: *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*. pp 3454–3461
19. He D, Yang X, Liang C, et al (2017) Multi-scale FCN with Cascaded Instance Aware Segmentation for Arbitrary Oriented Word Spotting In The Wild. In:

- Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp 3519–3528
20. Liao M, Shi B, Bai X, et al (2017) TextBoxes: A Fast Text Detector with a Single Deep Neural Network. 31st AAAI Conference on Artificial Intelligence, AAAI 2017 4161–4167
 21. Liao M, Zhu Z, Shi B, et al (2018) Rotation-Sensitive Regression for Oriented Scene Text Detection. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp 5909–5918
 22. Zhou Y, Ye Q, Qiu Q, Jiao J (2017) Oriented response networks. Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017 2017-Janua:4961–4970. <https://doi.org/10.1109/CVPR.2017.527>
 23. Long S, Ruan J, Zhang W, et al (2018) TextSnake: A Flexible Representation for Detecting Text of Arbitrary Shapes. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp 19–35
 24. Baek Y, Lee B, Han D, et al (2019) Character region awareness for text detection. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp 9357–9366
 25. Ye J, Chen Z, Liu J, Du B (2020) TextFuseNet: Scene text detection with richer fused features. IJCAI International Joint Conference on Artificial Intelligence 516–522. <https://doi.org/10.24963/ijcai.2020/72>
 26. Kim D, Kwak M, Won E, et al (2020) TLGAN: document Text Localization using Generative Adversarial Nets. In: arXiv:2010.11547v1 [cs.CV]
 27. Tian Z, Huang W, He T, et al (2016) Detecting text in natural image with connectionist text proposal network. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9912 LNCS:56–72. https://doi.org/10.1007/978-3-319-46484-8_4
 28. Li H, Wang P, Shen C (2017) Towards End-to-End Text Spotting with Convolutional Recurrent Neural Networks. Proceedings of the IEEE International Conference on Computer Vision 2017-Octob:5248–5256. <https://doi.org/10.1109/ICCV.2017.560>
 29. He T, Tian Z, Huang W, et al (2018) An End-to-End TextSpotter with Explicit Alignment and Attention. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. IEEE, pp 5020–5029
 30. Liu X, Liang D, Yan S, et al (2018) FOTS: Fast Oriented Text Spotting with a Unified Network. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 5676–5685. <https://doi.org/10.1109/CVPR.2018.00595>
 31. Deng D, Liu H, Li X, Cai D (2018) PixelLink: Detecting scene text via instance segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence
 32. Feng W, He W, Yin F, et al (2019) TextDragon: An End-to-End Framework for Arbitrary Shaped Text Spotting. In: 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, pp 9075–9084

33. Graves A, Fernández S, Gomez F, Schmidhuber J (2006) Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. *ACM International Conference Proceeding Series* 148:369–376. <https://doi.org/10.1145/1143844.1143891>
34. Liao M, Pang G, Huang J, et al (2020) Mask TextSpotter v3: Segmentation Proposal Network for Robust Scene Text Spotting. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. pp 706–722
35. Liao M, Wan Z, Yao C, et al (2020) Real-time scene text detection with differentiable binarization. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence* 11474–11481. <https://doi.org/10.1609/aaai.v34i07.6812>
36. Gupta A, Vedaldi A, Zisserman A (2016) Synthetic Data for Text Localisation in Natural Images. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, pp 2315–2324
37. Nguyen KC, Nguyen CT, Nakagawa M (2020) A Semantic Segmentation-based Method for Handwritten Japanese Text Recognition. *Proceedings of International Conference on Frontiers in Handwriting Recognition, ICFHR 2020-Sept*:127–132. <https://doi.org/10.1109/ICFHR2020.2020.00033>
38. Peng D, Jin L, Wu Y, et al (2019) A Fast and Accurate Fully Convolutional Network for End-to-End Handwritten Chinese Text Segmentation and Recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 25–30*. <https://doi.org/10.1109/ICDAR.2019.00014>
39. Michael J, Labahn R, Gruning T, Zollner J (2019) Evaluating sequence-to-sequence models for handwritten text recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 1286–1293*. <https://doi.org/10.1109/ICDAR.2019.00208>
40. Wang T, Wu DJ, Coates A, Ng AY (2012) End-to-end text recognition with convolutional neural networks. *Proceedings - International Conference on Pattern Recognition* 3304–3308
41. Smith R (2007) An Overview of the Tesseract OCR Engine. In: *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*. IEEE, pp 629–633
42. Wang K, Babenko B, Belongie S (2011) End-to-end scene text recognition. *Proceedings of the IEEE International Conference on Computer Vision* 1457–1464. <https://doi.org/10.1109/ICCV.2011.6126402>
43. Shi B, Bai X, Yao C (2017) An End-to-End Trainable Neural Network for Image-Based Sequence Recognition and Its Application to Scene Text Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39:2298–2304. <https://doi.org/10.1109/TPAMI.2016.2646371>
44. Wang J, Hu X (2017) Gated recurrent convolution neural network for OCR. In: *31st Conference on Neural Information Processing Systems*. pp 335–344
45. Sheng F, Chen Z, Xu B (2019) NRTR: A no-recurrence sequence-to-sequence model for scene text recognition. *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 781–786*. <https://doi.org/10.1109/ICDAR.2019.00130>

46. Lee J, Park S, Baek J, et al (2020) On recognizing texts of arbitrary shapes with 2D self-attention. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops 2020-June:2326–2335. <https://doi.org/10.1109/CVPRW50498.2020.00281>
47. Li M, Lv T, Cui L, et al (2021) TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models. In: arXiv:2109.10282v3 [cs.CL]
48. Wagner RC (1931) BEiT: BERT Pre-Training of Image Transformers. 465:1931–1932
49. Ation SL, Cl CS (2020) RoBERTa: A Robustly Optimized BERT Pretraining Approach. In: arXiv:1907.11692 [cs.CL]. pp 1–15
50. Wan Z, He M, Chen H, et al (2020) TextScanner: Reading characters in order for robust scene text recognition. AAAI 2020 - 34th AAAI Conference on Artificial Intelligence 12120–12127. <https://doi.org/10.1609/aaai.v34i07.6891>
51. Yu D, Li X, Zhang C, et al (2020) Towards accurate scene text recognition with semantic reasoning networks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2:12110–12119. <https://doi.org/10.1109/CVPR42600.2020.01213>
52. Ullah R, Sohani A, Rai A, et al (2018) OCR Engine to Extract Food-Items, Prices, Quantity, Units from Receipt Images, Heuristics Rules Based Approach. International Journal of Scientific & Engineering Research 9:1334–1341
53. Zhan F, Lu S (2019) ESIR: End-to-end scene text recognition via iterative image rectification. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2019-June:2054–2063. <https://doi.org/10.1109/CVPR.2019.00216>
54. Shi B, Yang M, Wang X, et al (2018) ASTER: An Attentional Scene Text Recognizer with Flexible Rectification. IEEE Transactions on Pattern Analysis and Machine Intelligence PP:1. <https://doi.org/10.1109/TPAMI.2018.2848939>
55. Sohani A, Ullah R, Ali F, et al (2019) Optical Character Recognition Engine to extract Food-items and Prices from Grocery Receipt Images via Templating and Dictionary-Traversal Technique. KIET Journal of Computing & Information Sciences [KJCIS] 2:59–73
56. Zhang Y, Chan W, Jaitly N Very Deep Convolutional Networks for End-to-End Speech Recognition. In: arXiv:1610.03022 [cs.CL]. pp 10–14
57. He Y, Chen C, Zhang J, et al (2021) Visual Semantics Allow for Textual Reasoning Better in Scene Text Recognition. In: arXiv:2112.12916 [cs.CV]
58. Guo H, Qin X, Liu J, et al (2019) EATEN: Entity-aware attention for single shot visual text extraction. Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 254–259. <https://doi.org/10.1109/ICDAR.2019.00049>
59. Mande R, Yelavarti KC, Jayalakshmi G (2018) Regular expression rule-based algorithm for multiple documents key information extraction. Proceedings of the International Conference on Smart Systems and Inventive Technology, ICSSIT 2018 262–265. <https://doi.org/10.1109/ICSSIT.2018.8748764>

60. Dhakal P, Munikar M, Dahal B (2019) One-Shot Template Matching for Automatic Document Data Capture. International Conference on Artificial Intelligence for Transforming Business and Society, AITB 2019 1:1–6. <https://doi.org/10.1109/AITB48515.2019.8947440>
61. D’Andecy VP, Hartmann E, Rusinol M (2018) Field extraction by hybrid incremental and a-priori structural templates. Proceedings - 13th IAPR International Workshop on Document Analysis Systems, DAS 2018 251–256. <https://doi.org/10.1109/DAS.2018.29>
62. Katti AR, Reisswig C, Guder C, et al (2020) Chargrid: Towards understanding 2D documents. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018. pp 4459–4469
63. Denk TI, Reisswig C (2019) BERTgrid: Contextualized Embedding for 2D Document Representation and Understanding. In: Workshop on Document Intelligence at NeurIPS 2019
64. Xu Y, Li M, Cui L, et al (2020) LayoutLM: Pre-training of Text and Layout for Document Image Understanding. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining 1192–1200. <https://doi.org/10.1145/3394486.3403172>
65. Sage C, Douzon T, Aussem A, et al (2021) Data-Efficient Information Extraction from Documents with Pre-trained Language Models. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 12917 LNCS:455–469. https://doi.org/10.1007/978-3-030-86159-9_33
66. Chua FC, Duffy NP (2021) DeepCPCFG: Deep Learning and Context Free Grammars for End-to-End Information Extraction. In: arXiv:2103.05908v2 [cs.CL]
67. Xu Y, Xu Y, Lv T, et al (2021) LayoutLMv2: Multi-modal Pre-training for Visually-rich Document Understanding. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). Association for Computational Linguistics, Stroudsburg, PA, USA, pp 2579–2591
68. Hwang W, Yim J, Park S, et al (2021) Spatial Dependency Parsing for Semi-Structured Document Information Extraction. In: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021. Association for Computational Linguistics, Stroudsburg, PA, USA, pp 330–343
69. Hong T, Kim D, Ji M, et al (2021) BROS: A Pre-trained Language Model Focusing on Text and Layout for Better Key Information Extraction from Documents. In: arXiv:2108.04539v4 [cs.CL]
70. Liu X, Gao F, Zhang Q, Zhao H (2019) Graph Convolution for Multimodal Information Extraction from Visually Rich Documents. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers). Association for Computational Linguistics, Stroudsburg, PA, USA, pp 32–39

71. Yu W, Lu N, Qi X, et al (2021) PICK: Processing Key Information Extraction from Documents using Improved Graph Learning-Convolutional Networks. In: 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, pp 4363–4370
72. Dang TAN, Thanh DN (2020) End-to-end information extraction by character-level embedding and multi-stage attentional u-net. 30th British Machine Vision Conference 2019, BMVC 2019
73. Wang J, Wang T, Tang G, et al (2021) Tag, Copy or Predict: A Unified Weakly-Supervised Learning Framework for Visual Information Extraction using Sequences. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence. International Joint Conferences on Artificial Intelligence Organization, California, pp 1082–1090
74. Zhang P, Xu Y, Cheng Z, et al (2020) TRIE: End-to-End Text Reading and Information Extraction for Document Understanding. MM 2020 - Proceedings of the 28th ACM International Conference on Multimedia 1413–1422. <https://doi.org/10.1145/3394171.3413900>
75. Huang Z, Chen K, He J, et al (2019) ICDAR2019 competition on scanned receipt OCR and information extraction. Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 1516–1520. <https://doi.org/10.1109/ICDAR.2019.00244>
76. Park S, Lee H (2019) CORD : A Consolidated Receipt Dataset for Post-OCR Parsing. In: Workshop on Document Intelligence at NeurIPS 2019