

Review

# A Survey of Graphical Page Object Detection with Deep Neural Networks

Jwalin Bhatt <sup>1,2,†</sup>, Khurram Azeem Hashmi <sup>1,2,3,\*,†</sup> , Muhammad Zeshan Afzal <sup>1,2,3,†</sup>  and Didier Stricker <sup>1,3</sup>

- <sup>1</sup> Department of Computer Science, Technical University, 67663 Kaiserslautern, Germany; jbhattach@rhrk.uni-kl.de (J.B.); muhammad\_zeshan.afzal@dfki.de (M.Z.A.); didier.stricker@dfki.de (D.S.)  
<sup>2</sup> Mindgrage, Technical University, 67663 Kaiserslautern, Germany  
<sup>3</sup> German Research Institute for Artificial Intelligence (DFKI), 67663 Kaiserslautern, Germany  
\* Correspondence: khurram\_azeem.hashmi@dfki.de  
† These authors contributed equally to this work.

**Abstract:** In any document, graphical elements like tables, figures, and formulas contain essential information. The processing and interpretation of such information require specialized algorithms. Off-the-shelf OCR components cannot process this information reliably. Therefore, an essential step in document analysis pipelines is to detect these graphical components. It leads to a high-level conceptual understanding of the documents that make the digitization of documents viable. Since the advent of deep learning, deep learning-based object detection performance has improved many folds. This work outlines and summarizes the deep learning approaches for detecting graphical page objects in document images. Therefore, we discuss the most relevant deep learning-based approaches and state-of-the-art graphical page object detection in document images. This work provides a comprehensive understanding of the current state-of-the-art and related challenges. Furthermore, we discuss leading datasets along with the quantitative evaluation. Moreover, it discusses briefly the promising directions that can be utilized for further improvements.

**Keywords:** deep neural network; document images; review paper; deep learning; performance evaluation; page object detection; graphical page objects; document image analysis; page segmentation



**Citation:** Bhatt, J.; Hashmi, K.A.; Afzal, M.Z.; Stricker, D. A Survey of Graphical Page Object Detection with Deep Neural Networks. *Appl. Sci.* **2021**, *11*, 5344. <https://doi.org/10.3390/app11125344>

Academic Editor: Sungho Kim

Received: 25 April 2021

Accepted: 2 June 2021

Published: 9 June 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The rapid increase in digitization of document images in both financial and non-financial sectors has considerably improved the accessibility of the data. To obtain reliable information from these scanned document images, options like manual capturing of data have become highly laborious and impractical. Therefore, over the last few decades, accurate information extraction has been vital research for the document analysis community [1–4].

Apart from the text, information in scanned documents is often stored in a graphical manner, such as tables, formulas, and figures. These are referred to as graphical page objects in document analysis community [5]. Figure 1 illustrates the problem that involves the detection of figures, formulas, and tables in document images. It is imperative to detect the graphical page objects before applying optical character recognition (OCR). One such scenario is information extraction from document images. Figure 2 illustrates the necessity of applying graphical page object detection systems for information extraction in document images. It is evident that even the state-of-the-art OCR method [6] fails to extract precise information from figures, tables, and formulas. Another application of such page object detection methods is document retrieval systems [7,8], where a document image having a specific type of page object is required. Therefore, it is essential to develop approaches that can parse the information from these page objects.

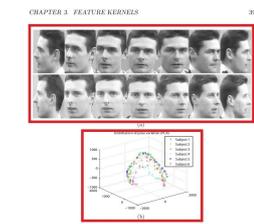


Figure 1.1. (A) A collection of images of two subjects with PCA solution and (B) the distribution of the first three principal components in a PCA subspace for all subjects.

years in the data [8].  
The kernel trick was first used by Aizerman, et al. [9] and Boser, et al. [3] to represent the input data to a system as nonlinear features. The input space is mapped by means of a nonlinear function, exponential or dot products, to a one-variant feature space in which the input data is nonlinearly related. The kernel trick created linearly mapped in the machine learning community, except with Support Vector Machine (SVM) [10], and Schölkopf et al. [20] developed Kernel Principal Component Analysis (KPCA). It was shown that KPCA is a linear invariant nonlinear feature and therefore provides better recognition performance over PCA. However, KPCA, just like PCA, extracts features along the direction of greatest variance across all the training samples and can not retain invariant

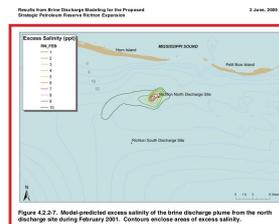


Figure 4.2.2.2. Model-predicted access ability of the beta discharge phase from the north discharge during February 2019. Contours enclose areas of access ability.

in [20] [21]. A formal definition of the contours should be provided in order of their order in which the contours occur. In addition, a set of axes which are not linearly related to the original axes should be provided. The contours should be provided in order of their order in which the contours occur. In addition, a set of axes which are not linearly related to the original axes should be provided. The contours should be provided in order of their order in which the contours occur. In addition, a set of axes which are not linearly related to the original axes should be provided.

2.3 Correction of Reviews

The area size of an informative review depends completely on the processing of input requests forwarded by users of the system. These input requests should result in an information system which reflects the modified structure of documents in a correct way.  
An informative review reflects an information system if and only if there exists a correspondence between the main and transitions of the application world and the state and transitions of the software of documents (DWS). From this it follows that the requirements for the software of documents are: the software of documents should be able to work in the context of the software of documents. This means that the software of documents should be able to work in the context of the software of documents. This means that the software of documents should be able to work in the context of the software of documents.



Figure 2: Processing of the User Request.

In [20] [21] a formal definition of the contours should be provided in order of their order in which the contours occur. In addition, a set of axes which are not linearly related to the original axes should be provided. The contours should be provided in order of their order in which the contours occur. In addition, a set of axes which are not linearly related to the original axes should be provided.

10.6 Factoring and Computing Euler's Phi Function

$p \leq M$ , which by the analysis in 4.7.6, happens with probability precisely

$$\prod_{p \leq M} (1 - \frac{1}{p}) = \frac{1}{M} \prod_{p \leq M} (1 - \frac{1}{p})$$

where

$$U(M) = \prod_{p \leq M} (1 - \frac{1}{p})$$

Now, the probability that this loop iteration produces an output is equal to the probability that the value  $n$  and  $\phi(n) \leq n$ , which is

$$\frac{U(M)}{M} = \prod_{p \leq M} (1 - \frac{1}{p})$$

Thus, every  $n$  is equally likely, and summing over  $n = 1, \dots, M$ , we see that the probability that this loop iteration succeeds in producing some output is  $U(M)$ .

Now consider the expected running time of this loop iteration. From the analysis in 4.7.6, it is easy to see that this is  $O(M^2)$ . That completes the analysis of a single loop iteration.

Finally, consider the behavior of Algorithm RFN as a whole. From our analysis of an individual loop iteration, it is clear that the output distribution of Algorithm RFN is as required, and if  $H$  denotes the number of loop iterations of the algorithm, then  $E[H] = U(M)^{-1}$ , which by Mertens' theorem is  $O(M)$ . Since the expected running time of each individual loop iteration is  $O(M^2)$ , it follows that the expected total running time is  $O(M^3)$ .

7.1.1 Using a probabilistic primality test

Analogous to the discussion in 5.7.1, we can analyze the behavior of Algorithm RFN under the assumption that  $n$  is a probabilistic algorithm which more or less randomly tests if a composite number is prime with probability bounded by  $\epsilon$ . Here, we assume that  $W_n$  denotes the expected running time of the primality test on input  $n$ , and set  $W_n = \text{max}\{W_1, \dots, W_M\}$ .

The situation here is a bit more complicated than in the case of Algorithm RPN, since an erroneous output of the primality test in Algorithm RFN could lead either to the algorithm halting prematurely (with a wrong output), or to the algorithm being delayed (because an opportunity to halt may be missed).

Let us first analyze in detail the behavior of a single iteration of the main loop of Algorithm RFN. Let  $A$  denote the event that the primality test makes a mistake in this loop iteration, and let  $\theta = P(A)$ . If  $T$  is the

4.2

efficient algorithm to solve one problem, we can efficiently solve the other, and vice versa.

Clearly, one direction is easy: if  $n$  can factor  $n$  into primes, so

$$n = p_1^{a_1} \dots p_k^{a_k} \quad (4.8)$$

then we can simply compute  $\phi(n)$  using the formula

$$\phi(n) = n \prod_{p|n} (1 - \frac{1}{p}) \quad (4.9)$$

For the other direction, first consider the special case where  $n = pq$ , for distinct primes  $p$  and  $q$ . Suppose we are given  $n$  and  $\phi(n)$ , so that we have two equations in the unknowns  $p$  and  $q$ :

$$n = pq \quad \text{and} \quad \phi(n) = (p-1)(q-1) \quad (4.10)$$

Substituting  $n/p$  for  $q$  in the second equation, and simplifying, we obtain

$$p^2 + \phi(n) - n = (p-1)n \quad (4.11)$$

which can be solved using the quadratic formula.

For the general case, it is just as easy to prove a stronger result: given any nonzero multiple of the exponent of  $\mathbb{Z}_n^*$ , we can efficiently factor  $n$ . In particular, this will show that we can efficiently factor Carmichael numbers.

Before stating the algorithm in full generality, we can convey the main idea by considering the special case where  $n = pq$ , where  $p$  and  $q$  are distinct primes, with  $p \equiv q \equiv 3 \pmod{4}$ . Suppose we are given  $n$  and  $\phi(n)$ , so that we have two equations in the unknowns  $p$  and  $q$ :

$$n = pq \quad \text{and} \quad \phi(n) = (p-1)(q-1) \quad (4.12)$$

Substituting  $n/p$  for  $q$  in the second equation, and simplifying, we obtain

$$p^2 + \phi(n) - n = (p-1)n \quad (4.13)$$

which can be solved using the quadratic formula.

The assumption that  $p \equiv q \equiv 3 \pmod{4}$  means that  $(p-1)/2$  is an odd integer, and since  $q$  is a multiple of  $p-1$ , it follows that  $q/(p-1) \equiv 1 \pmod{2}$ , and hence the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is the subgroup  $\mathbb{Z}_n^*$  of order 2, which is  $\{1, -1\}$ . Likewise, the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is  $\{1, -1\}$ . Let  $Z_n \times Z_n = Z_n \times Z_n$  be the ring isomorphism from the Chinese remainder theorem. Now, if  $\theta$  is the above algorithm does not lie in  $\mathbb{Z}_n^*$ , then certainly  $\text{ord}(\theta) \equiv 1 \pmod{2}$ . Otherwise, condition on the event that  $\theta \in \mathbb{Z}_n^*$ . In this conditional probability distribution,  $\theta$  is uniformly distributed over  $\mathbb{Z}_n^*$ , and if  $\theta \equiv \alpha^n \pmod{n}$  is uniformly distributed over  $\theta \in \mathbb{Z}_n^*$ . Let us consider each of these four possibilities:

- $\theta = 1$  implies  $\beta + 1 = (p+2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;
- $\theta = -1$  implies  $\beta + 1 = (p-2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;

4.2 Subcarrier Assignment Algorithm

The objective of the subcarrier assignment algorithm is to find the subcarrier assignment that maximizes the total

rate. This can be achieved if the subcarrier assignment is used to assign every active user to his best  $s_i$  subcarriers. Such an assignment problem is NP-complete. In this section, we will present a heuristic algorithm that finds a near-optimal solution to the subcarrier assignment problem. In particular, we will present a heuristic algorithm that finds a near-optimal solution to the subcarrier assignment problem. In particular, we will present a heuristic algorithm that finds a near-optimal solution to the subcarrier assignment problem.

Let us first analyze in detail the behavior of a single iteration of the main loop of Algorithm RFN. Let  $A$  denote the event that the primality test makes a mistake in this loop iteration, and let  $\theta = P(A)$ . If  $T$  is the

efficient algorithm to solve one problem, we can efficiently solve the other, and vice versa.

Clearly, one direction is easy: if  $n$  can factor  $n$  into primes, so

$$n = p_1^{a_1} \dots p_k^{a_k} \quad (4.8)$$

then we can simply compute  $\phi(n)$  using the formula

$$\phi(n) = n \prod_{p|n} (1 - \frac{1}{p}) \quad (4.9)$$

For the other direction, first consider the special case where  $n = pq$ , for distinct primes  $p$  and  $q$ . Suppose we are given  $n$  and  $\phi(n)$ , so that we have two equations in the unknowns  $p$  and  $q$ :

$$n = pq \quad \text{and} \quad \phi(n) = (p-1)(q-1) \quad (4.10)$$

Substituting  $n/p$  for  $q$  in the second equation, and simplifying, we obtain

$$p^2 + \phi(n) - n = (p-1)n \quad (4.11)$$

which can be solved using the quadratic formula.

The assumption that  $p \equiv q \equiv 3 \pmod{4}$  means that  $(p-1)/2$  is an odd integer, and since  $q$  is a multiple of  $p-1$ , it follows that  $q/(p-1) \equiv 1 \pmod{2}$ , and hence the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is the subgroup  $\mathbb{Z}_n^*$  of order 2, which is  $\{1, -1\}$ . Likewise, the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is  $\{1, -1\}$ . Let  $Z_n \times Z_n = Z_n \times Z_n$  be the ring isomorphism from the Chinese remainder theorem. Now, if  $\theta$  is the above algorithm does not lie in  $\mathbb{Z}_n^*$ , then certainly  $\text{ord}(\theta) \equiv 1 \pmod{2}$ . Otherwise, condition on the event that  $\theta \in \mathbb{Z}_n^*$ . In this conditional probability distribution,  $\theta$  is uniformly distributed over  $\mathbb{Z}_n^*$ , and if  $\theta \equiv \alpha^n \pmod{n}$  is uniformly distributed over  $\theta \in \mathbb{Z}_n^*$ . Let us consider each of these four possibilities:

- $\theta = 1$  implies  $\beta + 1 = (p+2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;
- $\theta = -1$  implies  $\beta + 1 = (p-2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;

3.2.2 Near Field Model Application

This algorithm is used to simulate the near-field behavior of the beta discharge plasma from the north and south discharge channels. The discharge parameters used are listed in Table 3.2.2.1 and the ambient conditions used are listed in Table 3.2.2.2.

Part	Vert. Angle	Beam Angle	Number of Parts	Part Discharge Rate	Discharge Rate	Discharge Sat. Temp.
(M)	0°	0°	50	2.200	200	25

A series of simulations were run using the LM2 model to characterize plasma behavior and diffusion under varying current levels present at the Rfion discharge area. Three constant currents based on the 50° and 90° percentiles were used and each case used the simple discharge from the cathodes. Currents for the Rfion sites were extracted from the CH2D model.

Diagn.	Current	Current Density	Ambient Sat. Temp.	Ambient Temperature
0.0	1.54	0	0	0
10.0	1.54	0	31	20
15.0	1.54	0	31	20
15.0	1.54	0	31	20

3.2.4 Far Field Transport Model Application  
This algorithm is used to simulate the far-field behavior of the beta discharge plasma from the north and south discharge channels. The discharge parameters used are listed in Table 3.2.4.1 and the ambient conditions used are listed in Table 3.2.4.2.

3754	SUNITE	Tuesday, 26 August 2008
Institution	Grant Proposal	Value
Royal Bank Research Institute	Programme Subsidy	392,300.00
	Grant Development Allowance	174,200.00
	Research Personnel Allowance	1,284,900.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
St. Vincent's Institute of Medical Research	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
Monash Children's Research Institute	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
Monash Children's Research Institute	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
Monash Children's Research Institute	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
Monash Children's Research Institute	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00
Monash Children's Research Institute	Training Personnel Allowance	822,000.00
	Research Allowance	778,000.00
	Programme Subsidy	44,000.00
	Research Personnel Allowance	274,000.00
	NSRF14	27,000.00
	Research Personnel Allowance	274,000.00

3.2.4.2 Ambient Conditions Used in the LM2 Model  
This algorithm is used to simulate the near-field behavior of the beta discharge plasma from the north and south discharge channels. The discharge parameters used are listed in Table 3.2.4.1 and the ambient conditions used are listed in Table 3.2.4.2.

Diagn.	Current	Current Density	Ambient Sat. Temp.	Ambient Temperature
0.0	1.54	0	0	0
10.0	1.54	0	31	20
15.0	1.54	0	31	20
15.0	1.54	0	31	20

3.2.4.3 Ambient Conditions Used in the LM2 Model  
This algorithm is used to simulate the near-field behavior of the beta discharge plasma from the north and south discharge channels. The discharge parameters used are listed in Table 3.2.4.1 and the ambient conditions used are listed in Table 3.2.4.2.

3.2.4.4 Ambient Conditions Used in the LM2 Model  
This algorithm is used to simulate the near-field behavior of the beta discharge plasma from the north and south discharge channels. The discharge parameters used are listed in Table 3.2.4.1 and the ambient conditions used are listed in Table 3.2.4.2.

4.2 Subcarrier Assignment Algorithm

The objective of the subcarrier assignment algorithm is to find the subcarrier assignment that maximizes the total

rate. This can be achieved if the subcarrier assignment is used to assign every active user to his best  $s_i$  subcarriers. Such an assignment problem is NP-complete. In this section, we will present a heuristic algorithm that finds a near-optimal solution to the subcarrier assignment problem. In particular, we will present a heuristic algorithm that finds a near-optimal solution to the subcarrier assignment problem.

Let us first analyze in detail the behavior of a single iteration of the main loop of Algorithm RFN. Let  $A$  denote the event that the primality test makes a mistake in this loop iteration, and let  $\theta = P(A)$ . If  $T$  is the

efficient algorithm to solve one problem, we can efficiently solve the other, and vice versa.

Clearly, one direction is easy: if  $n$  can factor  $n$  into primes, so

$$n = p_1^{a_1} \dots p_k^{a_k} \quad (4.8)$$

then we can simply compute  $\phi(n)$  using the formula

$$\phi(n) = n \prod_{p|n} (1 - \frac{1}{p}) \quad (4.9)$$

For the other direction, first consider the special case where  $n = pq$ , for distinct primes  $p$  and  $q$ . Suppose we are given  $n$  and  $\phi(n)$ , so that we have two equations in the unknowns  $p$  and  $q$ :

$$n = pq \quad \text{and} \quad \phi(n) = (p-1)(q-1) \quad (4.10)$$

Substituting  $n/p$  for  $q$  in the second equation, and simplifying, we obtain

$$p^2 + \phi(n) - n = (p-1)n \quad (4.11)$$

which can be solved using the quadratic formula.

The assumption that  $p \equiv q \equiv 3 \pmod{4}$  means that  $(p-1)/2$  is an odd integer, and since  $q$  is a multiple of  $p-1$ , it follows that  $q/(p-1) \equiv 1 \pmod{2}$ , and hence the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is the subgroup  $\mathbb{Z}_n^*$  of order 2, which is  $\{1, -1\}$ . Likewise, the image of  $\mathbb{Z}_n^*$  under the  $n$ -power map is  $\{1, -1\}$ . Let  $Z_n \times Z_n = Z_n \times Z_n$  be the ring isomorphism from the Chinese remainder theorem. Now, if  $\theta$  is the above algorithm does not lie in  $\mathbb{Z}_n^*$ , then certainly  $\text{ord}(\theta) \equiv 1 \pmod{2}$ . Otherwise, condition on the event that  $\theta \in \mathbb{Z}_n^*$ . In this conditional probability distribution,  $\theta$  is uniformly distributed over  $\mathbb{Z}_n^*$ , and if  $\theta \equiv \alpha^n \pmod{n}$  is uniformly distributed over  $\theta \in \mathbb{Z}_n^*$ . Let us consider each of these four possibilities:

- $\theta = 1$  implies  $\beta + 1 = (p+2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;
- $\theta = -1$  implies  $\beta + 1 = (p-2)/2$ , and hence  $\text{ord}(\beta) \equiv 1 \pmod{2}$ ;

Figure 1. Demonstration of the problem of graphical page object detection in document images. First Row: figure detection in document images. Second Row: detection of single and multiple formulas in document images. Third Row: localization of tabular areas in document images. The samples are taken from the dataset of ICDAR-17 POD [9].

With the recent surge of deep learning-based object detection algorithms in computer vision [10–12], a considerable amount of methods are developed which have formulated the problem of detecting graphical page objects in document images as an object detection problem. Furthermore, several datasets consisting of thousands of annotated scanned document images are also published. Although the approaches leveraging these datasets have significantly improved over-the-art, a consolidated comparison among these approaches is missing.

In this survey paper, we have presented a thorough analysis of the recent state-of-the-art approaches that have approached the problem of graphical page object detection in

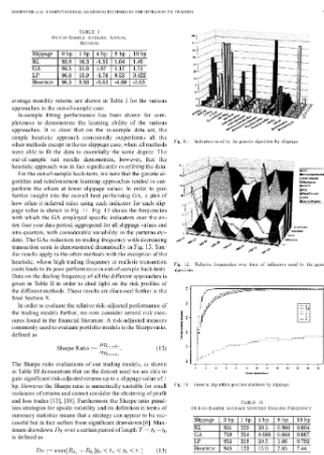
$i$	$s_i$	$s_{i+1}$	$s_{i+2}$	$s_{i+3}$	$s_{i+4}$	$s_{i+5}$	$s_{i+6}$	$s_{i+7}$	$s_{i+8}$	$s_{i+9}$
1	2	4	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
2	1	3	1	2	3	3	3	3	3	3
3	1	3	1	3	3	3	3	3	3	3
4	1	3	1	3	3	3	3	3	3	3
5	1	3	1	3	3	3	3	3	3	3

The estimate of  $\hat{s}_i$  is the average of all observations. At  $n=5$  we have  $\hat{s}_i > 2.6$ . To estimate  $\hat{s}_i$  at time  $n=5$ , we apply the PAVA to the sequence  $s_{i+1}, \dots, s_{i+n}$ . We see that the first violation of the order restriction occurs at  $n=2$ , and hence we replace the observations by the weighted average,  $(2.5 \times 2) / (2+2.5) = 2.25$ . This does not violate the first observation,  $s_{i+1}$ , since

$$2.25 < 2.5$$

9

scanned document images by employing deep neural networks. Since page objects can be of several types [13], we have covered the three most important page objects in document images [9]. These graphical page objects are referred to as table, formulas, and figures.



**Figure 2.** An illustration of information extraction through the conventional OCR method. On the left side of the figure, there is a document image containing multiple graphical page objects taken from the ICDAR-17 POD dataset [5], whereas the extracted information is present on the right side. We have applied an open-source Tesseract OCR [6] to extract the information. Since the OCR correctly recognized the textual content, we only demonstrate the extracted information from graphical page objects for brevity. The incorrectly extracted content depicts that graphical page object detection is an essential preliminary step before information extraction.

This paper investigates how deep neural network-based approaches work on detecting these types of page objects. Therefore, we have covered the most relevant approaches that have produced state-of-the-art results in this domain. Some of the discussed approaches work only on a single page object, and some have covered all three of them. However, our primary focus is to provide a perspective about the outcome of deep learning-based approaches on graphical page object detection in document images. To summarize, our contributions are as follows:

1. We present the comparisons between recently introduced algorithms for improving page object detection by highlighting their advantages and limitations.
2. We present a brief overview of the publicly available challenging datasets for graphical page object detection.
3. We provide an evaluative comparison among the state-of-the-art graphical page object detection systems.

Figure 3 illustrates the complete flow of this survey paper, whereas the remainder of the paper is organized as follows: Section 2 presents a brief overview of the prior works that have exploited traditional approaches to detect graphical page objects. Section 3 explains all the approaches contributing to graphical page objects by leveraging deep learning methods. Section 4 highlights all the publicly available datasets that can be employed to tackle the mentioned problem. Section 5 explains the mostly employed evaluation metrics and analyzes performances of all the discussed approaches in Section 3. Section 6 concludes the paper with a discussion on the current challenges and highlights the future directions.

DEMPSTER et al: COMPUTATIONAL LEARNING TECHNIQUES FOR INTRADAY FX TRADING 751

Slippage [0 bp | 1 bp | Top | Shp | IOP  
 RL 93.8 | 16.3 | -1.55 | 1.64 | 1.45  
 GA 94.5 | 21.6 | 1.67 | 1.17 | 1.71  
 LP 96.8 | 15.9 | -1.76 | 0.53 | 0.432  
 Heuristic [1] 96.3 | 8.56 | -5.03 | -4.89 | -5.63

LR, Sharpe Ratio :— (12)

FRearomta

Dr = max(R<sub>t</sub>, -R<sub>t</sub>) | 0 < ta < ty < 1 (13)

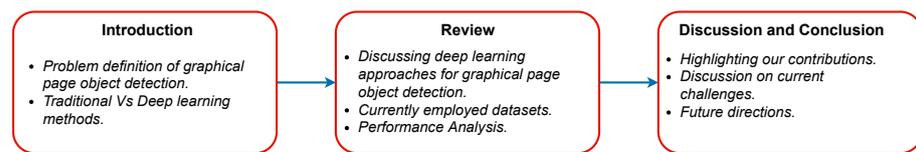
Fig. 11. Indicators used by the genetic algorithm by slippage.

Fig. 12 algorithm,

Fig. 13. Genetic algorithm position duration by slippage

TABLE II  
 'OUT-OF-SAMPLE AVERAGE MONTHLY DEALING FREQUENCY

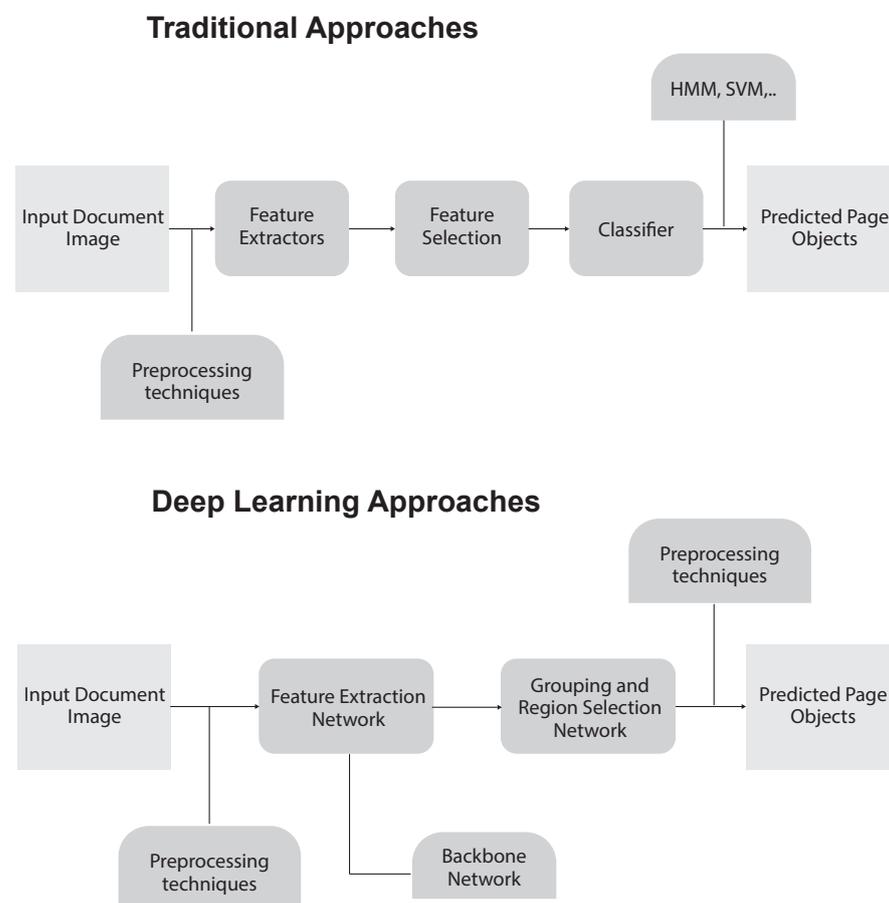
Slippage [1] 0 bp | 1 bp | 4bp | 8 bp | 10 bp  
 RL 851 | 220 | 20.5 | 0.980 | 0.604  
 GA 759 | 254 | 0.688 | 0.688 | 0.667  
 LP 852 | 218 | 20.5 | 1.06 | 0.792  
 Heuristic [1] 846 | 123 | 15.0 | 7.85 | 7.44



**Figure 3.** The block diagram illustrating the complete flow of the survey paper.

## 2. Traditional Approaches

The problem of graphical page object detection in documents is a well-recognized problem. Several approaches that employed traditional methods are introduced in this domain. Figure 4 illustrates the fundamental differences between the traditional approaches and the deep learning-based approaches. The traditional approaches leverage image processing techniques such as binarization and connected component analysis. Contrarily, deep learning-based methods utilize backbone CNN to generate the spatial feature maps from the document images.



**Figure 4.** Visual depiction of the basic differences between the traditional methods and the deep learning-based techniques. Traditional approaches rely heavily on image processing methods and custom heuristics whereas deep learning techniques leverage convolutional neural networks-based architectures. In deep learning approaches, spatial features from the document images are extracted from backbone networks such as VGG-16 [14] or ResNet [15]. These features are further propagated to region detection or segmentation networks to classify and localize page objects.

In order to implement table detection, the prior techniques [16–18] have defined a certain underlying structure for tables in a document. Tupaj et al. [19] employed Optical Character Recognition (OCR) to extract tabular information. The method tried to recognize possible table areas by analyzing the keywords and white spaces. The main disadvantage

of this approach is that it is fully based on the presumptions regarding the tables' structure and the collection of the used keywords.

Wang et al. [20] proposed another approach in the field of table analysis. It utilizes distance between consecutive words to detect table lines. Subsequently, adjacent vertical lines are grouped with consecutive horizontal words to propose table entity candidates. However, the underlying assumption is that there can be a maximum of two columns in a table. Hence, three types of layouts (single, double, and mixed columns) are designed in this approach. The drawback of this method is that it is only applicable to a limited number of designed templates.

Kieninger et al. [21–23] introduced a system called T-Recs to extract tabular information from documents. Their method takes the word bounding boxes which are segregated to build a segmentation graph in a bottom-up manner. Their system is vulnerable to tables containing multi rows and columns.

A method for detecting tables by calculating the intersection area between the vertical and horizontal lines was suggested by Gatos et al. [24]. The recreation of tables is then done by denoting corresponding vertical and horizontal lines related to intersection pairs. This approach presumes that a table should have ruling lines. A method for table detection by using Hidden Markov Models (HMMs) was suggested by Costa e Silva et al. [25]. The method fetches text from PDF files by applying the pdftotext Linux utility. Then feature vectors are computed based upon the gaps present between the text. This approach can only be employed for non-raster PDF files that do not contain noisy data.

A method for table detection under the assumption that tables in documents can contain only singular columns is proposed by Hu et al. [26]. Another technique for table detection in heterogeneous documents was proposed by Shafait et al. [18]. This mechanism is built into an open-source Tesseract OCR engine [6]. Although these traditional approaches were effective on the documents with restricted layout variations, either they rely on the meta-data or highly depends on the post-processing methods involving custom heuristics. Furthermore, they fail to produce similar results on generic datasets. Therefore, it is essential to exploit recently proposed deep learning techniques to tackle the problem of graphical page object detection in document images.

### 3. Methodologies

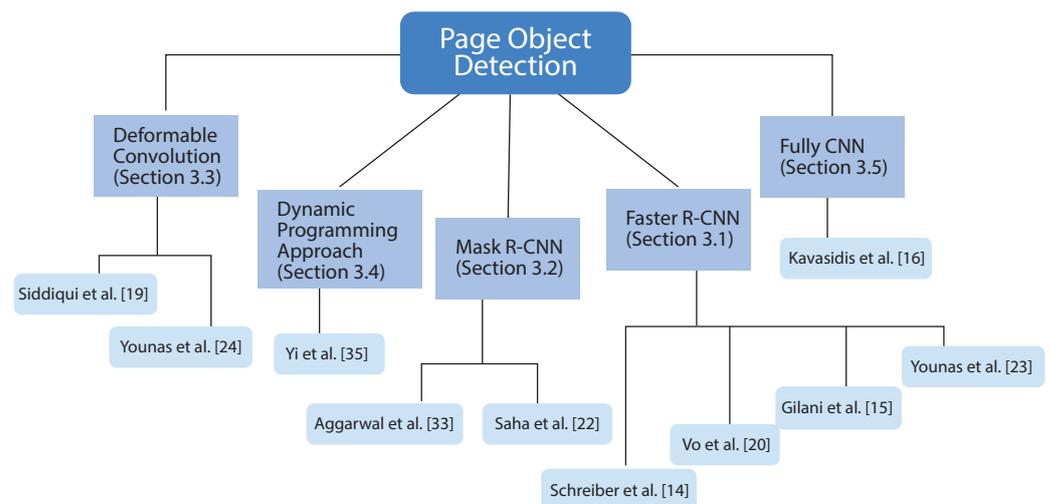
Graphical objects like tables, figures, and formulas are an integral part of documents because they hold a significant amount of information in a confined space. As explained in Section 1, detecting the graphical object means localizing these objects within a document image. Conceptually this problem is identical to localizing the objects in natural scene images. Recently, deep learning algorithms have also attracted the interest of researchers in the document image analysis community.

This section will discuss the methodologies that have utilized the capabilities of deep neural networks to solve the problem of graphical page object detection in document images. By following the convention of [5], we have covered approaches that have worked on the detection of the following graphical page objects in document images: (1) Tables, (2) Figures, and (3) Formulas.

For the convenience of our readers, we have classified the methodologies according to the employed deep learning concepts. We discuss the organizational flow of the methodologies in Figure 5. Table 1 summarizes the presented approaches and highlights their advantages and limitations.

**Table 1.** A Summary of different graphical page object detection methods that have employed deep neural networks.

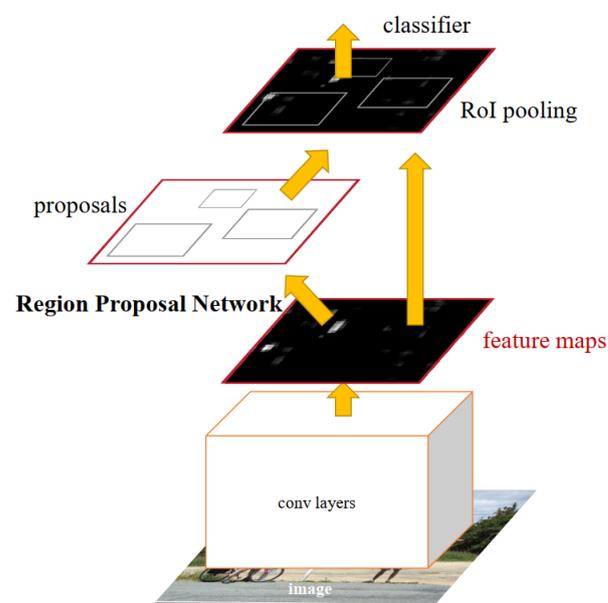
Literature	Method	Advantages	Disadvantages
De-CNT [27]	Deformable convolutions, implemented in the Faster R-CNN [11] architecture.	The dynamic receptive field of deformable CNN helps recognize tabular boundaries having arbitrary layouts.	Deformable CNN requires more computation as compared to conventional CNN.
Fi-Fo Detector [28]	Color transform, connected component analysis, distance transform applied on images that are fed to deformable pyramid network.	<b>(a)</b> Transformed images yield better results as compared to raw input images. <b>(b)</b> The approach leverages the deformable FPN model in their object detection network.	The approach depends on the extra pre-processing steps.
DeepDeSRT [29]	Faster R-CNN [11] with transfer learning.	Straightforward and effective approach to detect tables.	Does not perform as accurate as other state-of-the-art methods.
Vo et al. [30]	Ensembling of Fast R-CNN [31] and Faster R-CNN [11].	Leveraging the power of both selective search and region proposal network to generate reliable region of interests.	Computationally expensive because of combination of two separate object detection networks.
GOD [32]	Faster R-CNN [11] and Mask R-CNN [10].	Simple end-to-end approach to detect multiple page objects.	The network often mis-classifies the similar-looking page objects belonging to different classes.
Gilani et al. [33]	Transformed images are passed through Faster R-CNN [11].	The distance transform method helps the object detection network to focus on desired page object.	Requires extra pre-processing method.
CDeC-Net [34]	Cascaded Mask R-CNN [12].	<b>(a)</b> Multi-scale feature pyramid network. <b>(b)</b> Deformable Convolution improves the performance.	The method requires high computational resources due to composite backbone and deformable convolutions.
Kavasidis et al. [35]	Semantic image segmentation with saliency detection.	<b>(a)</b> Table detection formulated as a task of saliency detection. <b>(b)</b> Dilated convolutions instead of traditional convolution improve efficiency.	The approach depends on multiple pre-processing steps to achieve good results.
Yi et al. [36]	Dynamic programming based technique.	Replacing non maximal suppression with dynamic programming algorithm improves the refining process for region of interests.	Extra post-processing over head.



**Figure 5.** Categorization of discussed methodologies in the paper. The problem of page object detection is tackled through employing various deep learning concepts. The explained approaches are divided conceptually.

### 3.1. Faster R-CNN

Recently, it has been the case that the improvement of object detection algorithms in the field of computer vision has a direct relation with the improvement of graphical page object detection in document images. Faster R-CNN [11] which is the improved version of Fast R-CNN [31] is a two-stage object detection network. Figure 6 illustrates the architecture of Faster R-CNN. In order to obtain a detailed explanation about the architecture, readers may refer to [11]. This section covers the approaches that detect the graphical page objects by exploiting the capabilities of Faster R-CNN [11].



**Figure 6.** Explained architecture of Faster R-CNN. Image is obtained from [11].

An image-based deep learning table detection approach was suggested by Schreiber et al. [29] where they implemented Faster R-CNN for detection of tables in document images. The paper presents that the recently introduced object detectors dependent on Convolutional Neural Networks (CNN) can detect tables in document images. By leveraging back-bones like ZFNet [37] and VGG-16 [14], the authors have achieved promising results on ICDAR-13 dataset [38]. Their approach has also utilized the transfer learning technique

by using the pre-trained model on the Pascal-VOC dataset [39]. They also attempt table structure recognition along with table detection.

Vo et al. [30] published a method for page object detection, which involves detecting figures, formulas, and tables. Their technique makes use of an ensemble technique of Fast R-CNN [31], and Faster R-CNN [11]. They combined the region proposals obtained from Fast R-CNN and Faster R-CNN and then apply bounding box regression to boost performance. They have used the ICDAR-17 POD [5] dataset to benchmark their approach.

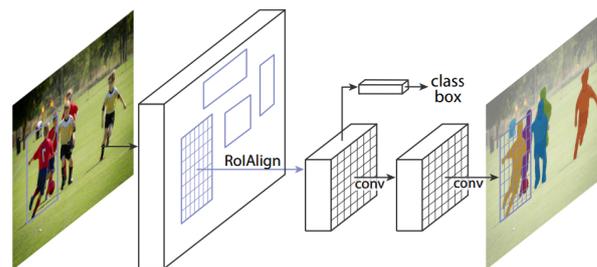
The blend of traditional methods and deep learning networks is presented by Younas et al. [40] to solve the problem of formula and figure detection in document images. The authors propose that instead of giving raw input images to object detection algorithms, transformed image representations yield better results. Connected component analysis (CC), distance transform, and color transform on the raw input images are performed and are subsequently processed using the Faster R-CNN model.

Gilani et al. [33] have utilized a similar technique. They have used the image transformation method in which a Euclidean distance transform [41], linear distance transform [42], and max distance transform [43] are applied on blue, green, and red channels of the input image, respectively. This transformed image is further propagated to Faster R-CNN to identify and regress the tabular boundaries in document images.

Another approach in which performance of two state-of-the-art object detection networks: Faster R-CNN [11] and Mask R-CNN [10] is compared on graphical page objects [32]. The article presents exhaustive evaluations on the detection of tables, formulas, and figures in document images. The paper's conclusion states that Mask R-CNN [10] is better suited to solve the problem of page object detection because of having extra components in the loss function.

### 3.2. Mask R-CNN

Mask R-CNN [10] is the extended model of Faster R-CNN [11] with an addition of an extra loss known as segmentation loss. Figure 7 depicts the basic architecture of Mask R-CNN. However, the comprehensive detail about the network can be found at [10]. The graphical page objects present in the document images have very low inter-class variance. An object originally labeled as a table can easily be misinterpreted with a figure or formula. By leveraging the segmentation loss of Mask R-CNN, researchers in the document image analysis community have improved the performance of graphical page object detection systems. This section covers those methodologies.



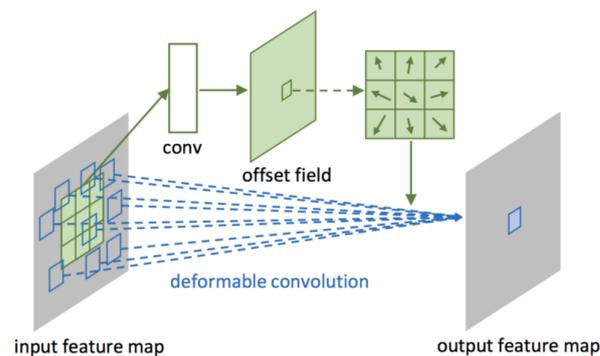
**Figure 7.** Explained architecture of Mask R-CNN. Image is obtained from [10].

Saha et al. [32] published the method for page object detection in document images through employing Mask R-CNN. Their end-to-end deep learning-based system, called Graphical Object Detection (GOD), detects the tables, figures, and formulas directly from the raw input images. The authors propose that there is no need to add extra pre or post-processing steps to solve page object detection. By leveraging the power of transfer learning, the authors have done benchmarking on the well-known datasets of ICDAR-17 POD [5], UNLV [44], and ICDAR-13 [38].

A recent end-to-end table detection network called CDeC-Net is introduced by Agarwal et al. [34]. The system CDeC-Net leverages the novel object detection network Cascade Mask R-CNN based on Cascade R-CNN [12]. The presented article has shown a noticeable improvement in the performance of table detection system across several datasets such as ICDAR-17 POD [5], ICDAR-13 [38], ICDAR-2019, Marmot [45], TableBank [46], PubLayNet [9], and UNLV [44]. After extensive evaluations, the authors have concluded that the network Cascade Mask R-CNN is superior to the previous state-of-the-art table detection systems.

### 3.3. Deformable Convolutions

Deformable convolutions differentiate the conventional convolutions by providing the leverage of deformable modules. The deformable module learns the sampling matrix with the location offsets. The offsets are learned according to the previous feature maps through additional convolution layers. This process makes the receptive field dynamic and enables the convolutional filters to adapt to different scales. While Figure 8 depicts the basic intuition behind the deformable convolutional networks, thorough information about the architecture is explained in [47]. Most of the mentioned methodologies have employed conventional convolutions in their object detection frameworks to solve page object detection in document images. Recently, instead of conventional convolutions, deformable convolutions [47] are investigated to detect tables, figures, and formulas. This section highlights those approaches.



**Figure 8.** Architecture of  $3 \times 3$  Deformable Convolution. Image is obtained from [47].

Siddiqui et al. [27] proposed an approach to detect tables that leverages deformable convolutions in their object detection framework. The authors argue that deformable convolutions are better suited for the problem of table detection. Because of their dynamic receptive field, tabular areas belonging to various scales and aspect ratios can be localized conveniently. The authors employed Faster R-CNN [11] by replacing a conventional Feature Pyramid Network (FPN) with a deformable FPN module. After extensive evaluations, the authors proved that deformable Faster R-CNN had outsmarted the conventional Faster R-CNN for the problem of table detection in document images.

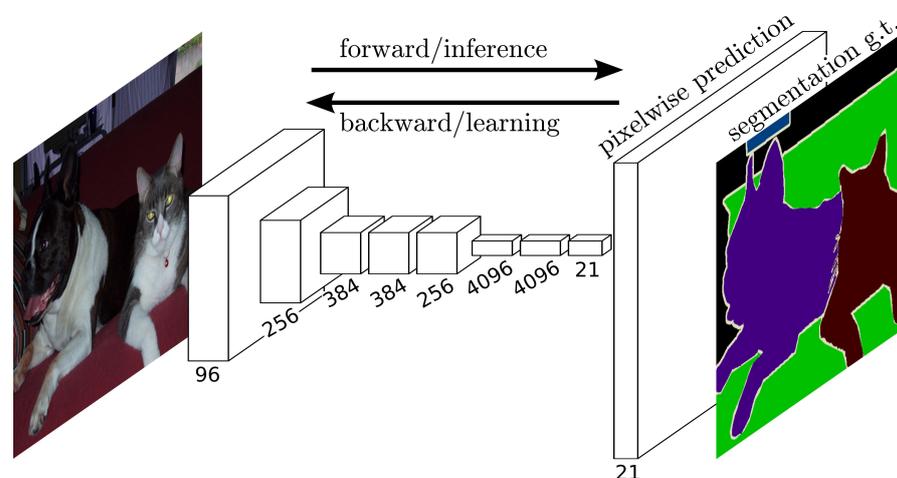
Younas et al. [28] exploited a similar approach by employing a deformable FPN module to detect formulas and figures in document images. Instead of providing raw input images to their deformable Faster R-CNN model, the authors have proposed an image transformation method identical to [40]. With the combination of transformed image representation and deformable object detection architecture, the authors have produced state-of-the-art results for the figure and formula detection on the famous ICDAR-17 POD dataset [5]. Along with the novel approach, the writers have also corrected the ICDAR-17 POD dataset [5] and have made it publicly available <https://bit.ly/2AUSlzl> (accessed on 25 April 2021).

### 3.4. Dynamic Programming Based Approach

Yi et al. [36] introduced a deep learning-based graphical page object detection approach similar to the object detection algorithms. In the presented approach, a convolutional neural network designed specifically for page object detection proposed candidate regions that are refined through a dynamic programming approach instead of the well-known non-maximum suppression method [48]. Tables, figures, formulas, and text lines are localized in document images by their system. The authors argue that page objects have a high variance in their aspect ratios, unlike objects in natural scenic images. Therefore, non-maximum suppression is not well-suited to detect all the page objects in a document image. The presented work compares the performance of their system with the conventional object detection approach of Fast R-CNN [31] and Faster R-CNN [11], and concludes that the dynamic programming-based approach has outperformed the rest of the methods.

### 3.5. Fully Convolutional Neural Networks

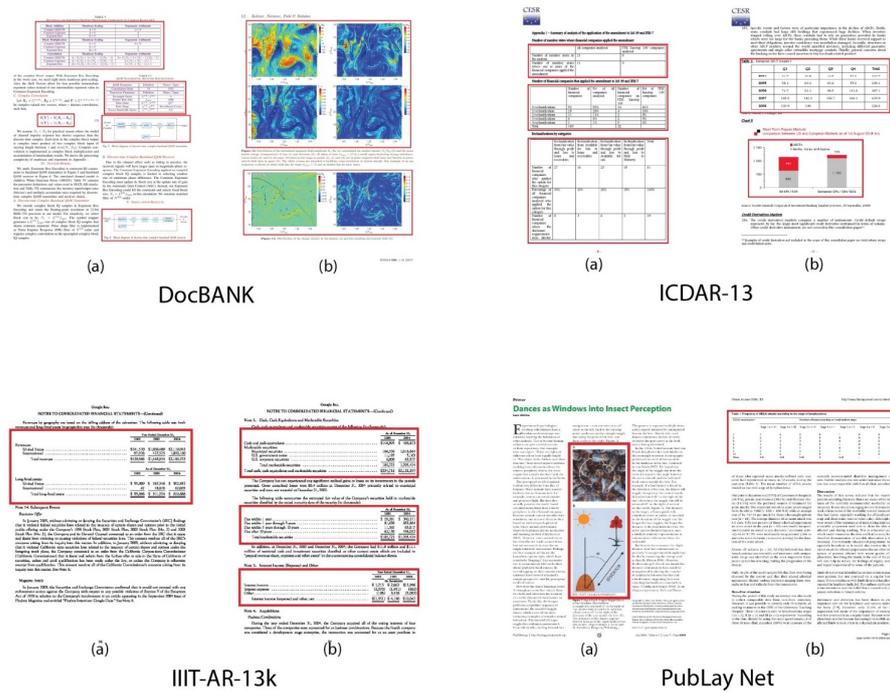
Along with object detection algorithms, Fully Convolutional Neural Networks (FCNNs) [49] have been exploited to solve graphical page object detection in document images. The basic intuition behind FCNNs is assigning the label for each pixel present in an image. Figure 9 depicts the architecture of FCNNs and for further explanation, we refer our readers to [49]. Kavasidis et al. [35] posed the problem of tables and chart detection as a saliency detection problem. The authors propose that each class of page object can be referred to as a separate saliency category. To segment those categories (tables and charts), the system employs FCNNs where each pixel will be classified into tables, charts, or a background in a document image. The obtained saliency map is further propagated to the fully connected Conditional Random Field (CRF) [49], which smooths the system's output.



**Figure 9.** Explained architecture of Fully Convolutional Neural Network. Image is obtained from [50].

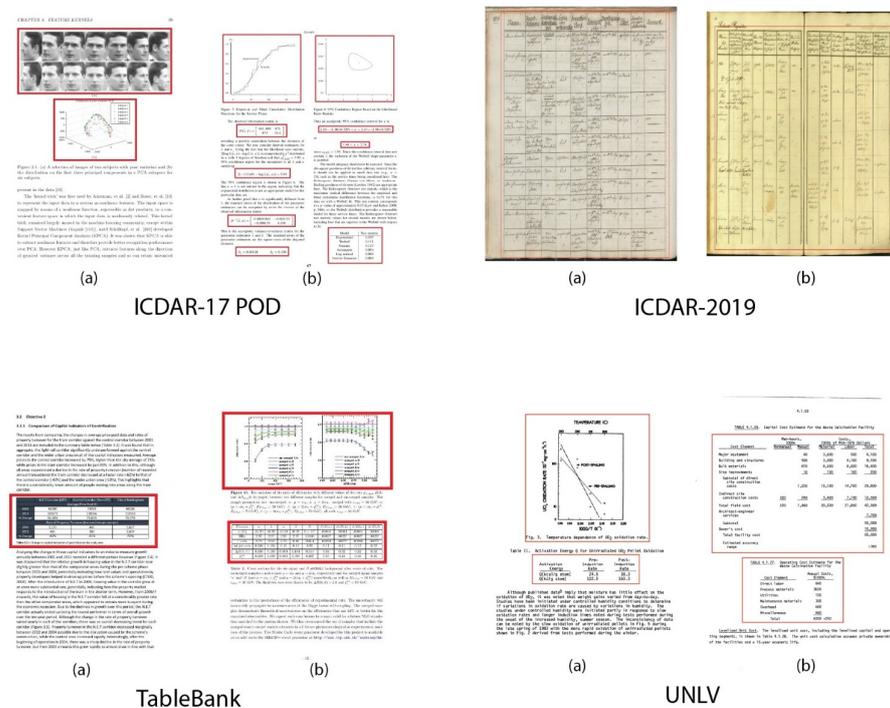
## 4. Datasets

Deep neural networks consist of a huge number of parameters. To achieve convergence, datasets with a massive amount of images are required to train these networks optimally [27,29]. Recently, the document image analysis community published several public datasets. Some of these datasets have provided annotations for various graphical page objects. This section will mainly cover the recently published datasets that contain information about the boundaries of tables, formulas, and figures. Figure 10 depicts few samples of these datasets.



**Figure 10.** Sample document images taken from the various datasets of DocBank [13], ICDAR-13 [38], IIT-AR-13K [51], and PubLayNet [9]. Part (a,b) represent the highlighted graphical page objects in a document image.

Moreover, we discuss few datasets that only contain annotations for one of the three mentioned page objects, such as tables. Figure 11 depicts a couple of samples belonging to these datasets. Table 2 presents the summary of all the datasets covered in this section.



**Figure 11.** Sample document images taken from the various datasets of ICDAR-17 POD [5], ICDAR-19 [52], TableBank [46], and UNLV [44]. Part (a,b) represent the highlighted graphical page objects in a document image. It is important to mention that most of the datasets illustrated in this figure have annotations for the tabular boundaries only.

**Table 2.** Graphical page object datasets. It is important to mention that we have considered equation and formula as semantically equal in this table. Some of these datasets contain as many as 12 page objects [13]. For the sake of convenience, we have only included table, figure, and formula.

Dataset	Table	Figure	Formula	# Samples	Year	Location
PubLayNet [9]	✓	✓	✗	360 K	2019	<a href="https://developer.ibm.com/exchanges/">https://developer.ibm.com/exchanges/</a> (accessed on 25 April 2021)
DocBank [13]	✓	✓	✓	500 K	2020	<a href="https://doc-analysis.github.io/docbank-page">https://doc-analysis.github.io/docbank-page</a> (accessed on 25 April 2021)
ICDAR-17 POD [5]	✓	✓	✓	2.4 K	2017	<a href="https://www.icst.pku.edu.cn/cpdp">https://www.icst.pku.edu.cn/cpdp</a> (accessed on 25 April 2021)
IIIT-AR-13k [51]	✓	✓	✗	13 K	2020	<a href="http://cvit.iiit.ac.in/usodi/iiitar13k.php">http://cvit.iiit.ac.in/usodi/iiitar13k.php</a> (accessed on 25 April 2021)
DeepFigures [53]	✓	✓	✗	5.5 K	2018	<a href="https://s3-us-west-2.amazonaws.com/ai2-s2-research-public">https://s3-us-west-2.amazonaws.com/ai2-s2-research-public</a> (accessed on 25 April 2021)
ICDAR-13 [38]	✓	✗	✗	238	2013	<a href="http://www.tamirhassan.com/html/">http://www.tamirhassan.com/html/</a> (accessed on 25 April 2021)
UNLV [44]	✓	✗	✗	427	2010	<a href="http://www.iapr-tc11.org/mediawiki/index.php?">http://www.iapr-tc11.org/mediawiki/index.php?</a> (accessed on 25 April 2021)
ICDAR-2019 [52]	✓	✗	✗	3.6 K	2019	<a href="https://zenodo.org/record/2649217">https://zenodo.org/record/2649217</a> (accessed on 25 April 2021)
Marmot [45]	✓	✗	✓	958	2012	<a href="https://www.icst.pku.edu.cn/cpdp/sjzy">https://www.icst.pku.edu.cn/cpdp/sjzy</a> (accessed on 25 April 2021)
TableBank [45]	✓	✗	✗	417 K	2020	<a href="https://doc-analysis.github.io/tablebank-page">https://doc-analysis.github.io/tablebank-page</a> (accessed on 25 April 2021)

#### 4.1. ICDAR-17 POD

ICDAR-17 Page Object Detection (POD) [5] is a publicly available dataset introduced in the page object detection competition at ICDAR 2017. This dataset is one of the most widely used datasets to evaluate graphical page object detection systems. The dataset comprises a page consisting of various layouts such as single-column, double-columns, and multi-columns. This dataset has an annotation for tables, formulas, figures present in document images. The page objects contain headings, textual, page title, content, captions, etc. This dataset contains 2417 English document images in total, extracted from 1500 scientific papers of CiteSeer. This dataset is split into training and test set, having 1600 and 817 document images, respectively. Contents are as follows: Training set contains 1978 figures, 698 tables, and 3515 formulas, while Test set contains 961 figures, 371 tables, and 1192 formulas.

#### 4.2. PubLayNet

In 2019, Zhong et al. [9] published a huge dataset for document layout analysis known as PubLayNet. This dataset is generated by automatically annotating the document layout of over 1 million PubMed Central™ PDF articles. The dataset contains various document layout categories, including text, title, list, table, and figures. Having more than 360 thousand annotated document images, this huge dataset facilitates the researchers to develop and evaluate advanced deep learning-based models for document page object detection.

#### 4.3. DocBank

Another dataset to solve document layout analysis is released by Li et al. [13]. The dataset is known as DocBank, which is the extended version of the TableBank dataset [46]. DocBank is a novel large-scale dataset, and it is constructed by employing weak supervision from the LaTeX documents available on arXiv.com. The proposed dataset comprises 500 thousand document pages with 12 different kinds of semantic blocks such as tables, figures, equations, figures, lists, paragraphs, etc. The authors also define the training/val/test splits in which 400 thousand samples are used for the training purpose, whereas 50 thousand samples are allocated for validation and testing purposes. This large-scale rich dataset extends the opportunities to investigate the blend of deep neural networks employed in computer vision with the methods mainly used in document analysis.

#### 4.4. Marmot

Marmot is widely utilized by scientists in the area of understanding the tables and formulas. This dataset has been published by the Institute of Computer Science and Technology (Peking University) and described in the paper proposed by Fang et al. [45]. This dataset consists of 2000 document images. These images are comprised of conference papers of both English and Chinese languages from 1970 to 2011. There is roughly a 1:1 ratio for both positive and negative images in the dataset. Due to the complex page layouts, this dataset is highly applicable for evaluating table detection systems. There were few instances of incorrect annotations in the dataset, which is corrected by Schreiber et al. [29]. The size of the dataset reduces to 1967 document images after the correction.

#### 4.5. TableBank

During early 2019, in the community dedicated to table detection, Minghao et al. [46] recognize the requirement for enormous datasets and established TableBank. TableBank is a dataset consisting of 417 thousand labeled images utilizing tabular data. The dataset has been accumulated by gathering the information over the documents which are present in .docx format. The dataset also contains another form of information, that is, Latex documents which were accumulated from the arXiv 5 database. The publishers of this dataset suggest the usage of this dataset for both structural recognition and table detection tasks. The authors of this dataset claim that this large-scale dataset will enable the researchers to exploit the capacity of deep neural networks.

#### 4.6. IIIT-AR-13k

Mondal et al. [51] have proposed a novel IIIT-AR-13k dataset. The dataset mainly consists of business-type documents. There are in total 13 thousand pages containing graphical elements like tables, signatures, figures, and so on. The bounding boxes are marked in a non-automated manner to construct the dataset. The authors generated this dataset manually, and it is one of the biggest datasets in the domain of graphical page object detection.

#### 4.7. DeepFigures

Based on our knowledge, DeepFigures [53] is one of the most extensive free-to-use datasets to utilize for the task of graphical page object detection. It comprises more than 1.4 million documents along with the information of boundaries of tables and figures. The authors leverage the scientific articles found online on the databases like PubMed and arXiv to create the dataset. This large-scale dataset provides an opportunity to investigate the performance of table and figure detection systems in document images.

#### 4.8. ICDAR-13

ICDAR-13 [38] is widely utilized for the problem of table detection and table structure extraction. The dataset consists of PDF files. These PDF files are converted into images. The dataset is composed of graphs, structured tables, text as information, and charts. However, It only provides annotations for structure data for table recognition and table detection. This dataset has 67 PDFs with 150 tables in which 27 PDFs are from the EU, and 40 PDFs are from the US Government. In total, this dataset has 238 images, from which 128 images contain table information. This dataset is often used for reporting and comparing.

#### 4.9. UNLV

For document image analysis, UNLV [44] is a very well-known dataset in this field. This dataset is composed of various documents like business letters, magazines, reports, newspapers, etc. Even though this document has almost 10,000 images, only 427 images possess a tabular region. Often, the research community uses the images that contain the tabular regions to manage numerous experiments.

#### 4.10. ICDAR-2019

In 2019, Competition on Table Detection and Recognition(cTDaR) [52] is executed in ICDAR. The competition proposes two new datasets: historical and modern datasets. The historical datasets encompass train schedules, simple tabular prints from old books, images from hand-written accounting ledgers, and so on. In contrast, modern datasets encompass samples from forms, financial documents, and scientific papers. This dataset has become a benchmark dataset to assess the performance of state-of-the-art systems for table analysis.

#### 5. Evaluation

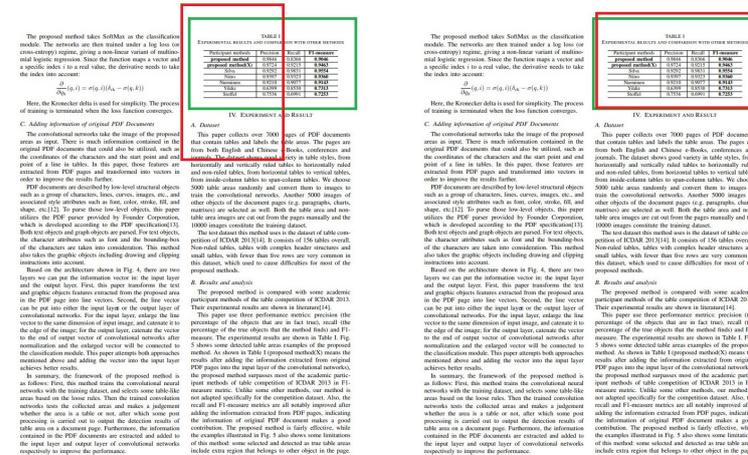
This section covers the well-known evaluation metrics that have been employed by the deep learning-based approaches to assess their performance and compares the results among various state-of-the-art approaches. Moreover, we will present the comprehensive evaluative comparison between the methodologies that are explained in Section 3.

##### 5.1. Precision

The metric precision is defined as the ratio between the correctly predicted positives samples to the total positive samples. Figure 12 depicts the definition of precision for the problem of graphical page objects in document images. Mathematically, it is described as:

$$Precision = \frac{\text{correct prediction}}{\text{total predictions}} = \frac{TP}{TP + FP} \tag{1}$$

where TP denotes the True Positives and FP represents False Positives.



**Figure 12.** An instance of a precise and imprecise table detection. Green color represents the ground-truth tabular area whereas red color denotes the predicted tabular boundary.

##### 5.2. Recall

The metrics recall evaluates the performance of the system by calculating the number of corrected predictions in the actual test set. It is calculated as follows:

$$Recall = \frac{\text{correct predictions}}{\text{Total correct annotations in ground-truth}} = \frac{TP}{TP + FN} \tag{2}$$

where TP denotes the True Positives and FN represents False Negatives.

##### 5.3. F-Measure

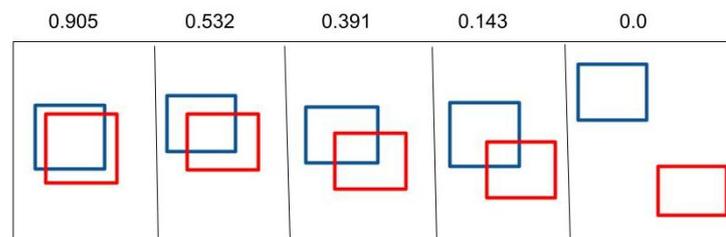
The harmonic mean of precision and recall is known as F-Measure. The formula for finding an F1 score is given by:

$$F\text{-Measur}e = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

#### 5.4. Intersection Over Union

Intersection Over Union (IOU) is a well-known evaluation metric commonly used in evaluating the capabilities of object detection algorithms. Since object detection techniques have been widely exploited to solve graphical page object detection, we have decided to discuss this metric in our paper. IOU calculates how much the area of the predicted bounding box intersects with the area of the actual ground-truth. For the sake of convenience, an example of computing IOU is illustrated in Figure 13. Mathematically, it is explained as follows:

$$IOU = \frac{\text{Area of Overlap region}}{\text{Area of Union region}} \quad (4)$$



**Figure 13.** Visual illustration of IOU in object detection methods. The bounding box with blue color represents the ground-truth whereas the bounding box with red color denotes the predicted bounding box. Considering the IOU threshold set to 0.5, only the first two predictions from the left will be considered true positives whereas the rest of them will be treated as false positives.

The problem of graphical page object detection is to localize the boundaries of formulas, figures, and tables. For the sake of visual convenience, we have divided the performance evaluation between the explained methodologies into three separate tables. The quantitative analysis for detection of tables, figures and formulas are summarized in Tables 3–5 respectively. Various approaches have evaluated their methods on distinctive IOU thresholds.

**Table 3.** Table detection performance evaluation across various datasets. The double horizontal line divides the employed datasets.

Literature	Year	Dataset	IOU	Precision	Recall	F-Measure	Method
Saha et al. [32]	2019	ICDAR-17 POD	0.6	-	-	0.971	Mask R-CNN
Siddiqui et al. [27]	2018	ICDAR-17 POD	0.6	0.965	0.971	0.968	Deformable Faster R-CNN
Agarwal et al. [34]	2020	ICDAR-13	0.5	1	1	1	CDEC-Net
Saha et al. [32]	2019	ICDAR-13	0.5	0.982	1	0.991	Mask R-CNN
Kavasidis et al. [35]	2018	ICDAR-13	0.5	0.981	0.981	0.981	Fully Convolutional Network
Siddiqui et al. [27]	2018	ICDAR-13	0.5	0.996	0.996	0.996	Deformable FPN
Schreiber et al. [29]	2017	ICDAR-13	0.5	0.974	0.962	0.968	Faster R-CNN
Agarwal et al. [34]	2020	UNLV	0.5	0.960	0.770	0.865	CDeC-Net
Saha et al. [32]	2019	UNLV	0.5	0.946	0.910	0.928	Mask R-CNN
Siddiqui et al. [27]	2018	UNLV	0.5	0.786	0.749	0.767	Deformable FPN
Gilani et al. [33]	2017	UNLV	0.5	0.823	0.907	0.863	Faster R-CNN
Agarwal et al. [34]	2020	ICDAR-2019	0.5	0.987	0.946	0.966	CDeC-Net

**Table 4.** Figure detection performance comparison.

Literature	Year	Dataset	IOU	Precision	Recall	F-Measure	Method
Younas et al. [28]	2020	ICDAR-17 POD	0.6	0.931	0.913	0.922	Fi-Fo with Deformable Convoltuions
Saha et al. [32]	2019	ICDAR-17 POD	0.6	-	-	0.918	Mask R-CNN

**Table 5.** Formula detection performance comparison.

Literature	Year	Dataset	IOU	Precision	Recall	F-Measure	Method
Younas et al. [28]	2020	ICDAR-17 POD	0.6	0.957	0.952	0.954	Fi-Fo with Deformable Convoltuions
Saha et al. [32]	2019	ICDAR-17 POD	0.6	-	-	0.924	Mask R-CNN

### 5.5. Evaluation for Table Detection

It can be observed by looking at Table 3, the instance segmentation-based architectures like Cascade Mask R-CNN has outperformed the rest of the approaches with a slight margin. It shows that the multi-scale classification module that has improved the generic object detection [54], has also advanced the table detection systems in document images.

### 5.6. Evaluation for Figure Detection

Table 4 compares the performance between the two recently proposed deep learning-based approaches for figure detection in document images. It is evident that the approach with deformable convolutions has outranked the instance segmentation-based approach. This is because of the dynamic receptive field that takes care of the figures having various scales and aspect ratios in the document images. The results also entail that instead of providing raw images to the deep neural network, transforming images through traditional document image analysis methods can yield better results.

### 5.7. Evaluations for Formula Detection

The performance assessment between the two novel approaches is explained in Table 5. Analogous to the figure detection, the approach with the blend of image transformations and deformable convolutions has out-smarted the other method for formula detection.

While evaluating page object detection systems, it is essential to mention that still there is a room for improvement to come up with deep neural networks that can localize and classify all the page objects present in a document image. So far, we have seen that particular methods or modules are utilized to detect various page objects.

## 6. Discussion and Conclusions

The process of extracting precise information from graphical page objects is a crucial and challenging problem in document image analysis and has received noticeable attention. The state-of-the-art page object detection systems have been remarkably improved due to recent advances in deep learning. This survey paper has provided a comprehensive overview of approaches that perform end-to-end graphical element detection in document images. Furthermore, this paper presents a structural taxonomy for the approaches according to the utilized deep learning method in Section 3. It compares these methods by highlighting their advantages and disadvantages in Table 1. Moreover, we explain the recently employed datasets in Section 4 and summarize their essential statistics in Table 2. Furthermore, we talk about the currently used evaluation criteria and analyze the performance of current deep learning based-graphical page object detection systems in Section 5. We conclude this survey paper with a discussion on the current difficulties and challenges in Section 6.1, and finally recommended some future directions in Section 6.2.

### 6.1. Difficulties and Challenges

After reviewing several methods in the field of graphical page object detection, we have noticed some key issues that deserve to be addressed. These are as follows:

- First and foremost is that the current state-of-the-art performs better when the network is trained for a single type of graphical object, i.e., only for table or only for formula. The performance of a graphical page object degrades when it is trained to detect multiple graphical page objects in document images [5,32].
- The second critical challenge is low inter-class (between different classes) and high intra-class (within the same class) variation. Due to low inter-class variance, tables without ruling lines can easily be misclassified with algorithms or mathematical formulas and vice-versa [27,55]. Similarly, a figure can be falsely predicted as a table and vice-versa on account of low intra-class variance.
- The datasets differ significantly from each other. At present, several datasets only focus on a single graphical page object [38,46,52]. Therefore, there is a growing need for large-scale datasets that provide annotations for multiple page objects like figures, formulas, and tables [5,13].
- The recent two-stage object detection networks are generally gigantic in size [56,57]. It is not easy to process the images at their original resolution with limited computational resources. Therefore, some important features are compromised during the downsampling process in the case of detecting smaller graphical page objects such as embedded formulas [58].
- Most of the current state-of-the-art methods rely on some post-processing to obtain reliable results [28]. Therefore, more generic deep learning-based solutions are required that can detect distinctive graphical page objects in a diverse environment.

Because of the challenges mentioned above, we can conclude that standardization is required with diversity to tune the methods towards generic graphical object detection in document images. Moreover, the development of methods tailored only for graphical page object detection in document images can significantly improve the performance. This work is one effort to unify the performance of the deep neural network architectures for most renowned datasets.

### 6.2. Future Work

There are many possibilities to explore in order to improve the performance of graphical page object detection in document images. In general, recently proposed novel neural network architectures for object detection [56,59–61] can improve performance of graphical page object detection systems. The second promising direction is the multimodal processing of the graphical objects. In the case of graphical page object detection, multimodal processing, in the simplest form, is the processing of image information and text information together [62,63]. An example of such a case is when a figure is categorized as a table and vice versa; the text information can be beneficial. The table is the most complicated graphical page object among all the graphical page objects [48]. To improve the performance further, another promising path to explore is the localization of individual columns and rows of the specified tables. Furthermore, identifying headers of the table can significantly help to understand the table's inner structure. Furthermore, the following directions can be explored in the future:

- **Weak/Unsupervised learning:** At present, all the reliable graphical page object detection systems depend on large-scale labeled datasets. The processing of annotating the document images with graphical page objects is laborious and inefficient. Hence, there is a dire need to build weak/unsupervised graphical page object detection systems that produce impressive results after training limited samples.
- **Light weight systems:** Modern state-of-the-art graphical page object detection methods are not efficient at all. However, there is a growing need to build intelligent information extraction systems that can work effortlessly on mobile devices [64,65].

- **Domain adaptation:** There is still a significant gap in developing clever page object detection methods that can adapt to different domains. An example of such a scenario is building a system that works equally well on the historical and modern document images.
- **Neural architecture search:** Deep learning enables us to eliminate custom features engineering, which demands domain knowledge. However, the current employed deep neural network also requires setting precise hyperparameters. Another exciting direction could be leveraging neural architectural search to automate the design of a number of layers and anchor settings as accomplished in the field of computer vision [66–69].

**Author Contributions:** Writing—original draft preparation, J.B., K.A.H., M.Z.A.; writing—review and editing, K.A.H., M.Z.A.; supervision and project administration, D.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** The work leading to this publication has been partially funded by the European project INFINITY under Grant Agreement ID 883293.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Mori, S.; Suen, C.Y.; Yamamoto, K. Historical review of OCR research and development. *Proc. IEEE* **1992**, *80*, 1029–1058. [[CrossRef](#)]
2. Breuel, T.M. The OCRopus open source OCR system. In *Document Recognition and Retrieval XV*; International Society for Optics and Photonics: Bellingham, WA, USA, 2008; Volume 6815.
3. Hashmi, K.A.; Ponnappa, R.B.; Bukhari, S.S.; Jenckel, M.; Dengel, A. Feedback Learning: Automating the Process of Correcting and Completing the Extracted Information. In Proceedings of the 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW), Sydney, Australia, 20–25 September 2019; Volume 5, pp. 116–121.
4. Pondenkandath, V.; Seuret, M.; Ingold, R.; Afzal, M.Z.; Liwicki, M. Exploiting state-of-the-art deep learning methods for document image analysis. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 5, pp. 30–35.
5. Gao, L.; Yi, X.; Jiang, Z.; Hao, L.; Tang, Z. ICDAR2017 competition on page object detection. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 1417–1422.
6. Smith, R. An overview of the Tesseract OCR engine. In Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Parana, Brazil, 23–26 September 2007; Volume 2, pp. 629–633.
7. Deveaud, R.; Mothe, J.; Ullah, M.Z.; Nie, J.Y. Learning to adaptively rank document retrieval system configurations. *ACM Trans. Inf. Syst.* **2018**, *37*, 1–41. [[CrossRef](#)]
8. Sharma, D.K.; Pamula, R.; Chauhan, D.S. A hybrid evolutionary algorithm based automatic query expansion for enhancing document retrieval system. *J. Ambient. Intell. Hum. Comput.* **2019**. [[CrossRef](#)]
9. Zhong, X.; Tang, J.; Yepes, A.J. Publaynet: Largest dataset ever for document layout analysis. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 1015–1022.
10. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
11. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv* **2015**, arXiv:1506.01497.
12. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High quality object detection and instance segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1483–1498. [[CrossRef](#)]
13. Li, M.; Xu, Y.; Cui, L.; Huang, S.; Wei, F.; Li, Z.; Zhou, M. Docbank: A benchmark dataset for document layout analysis. *arXiv* **2020**, arXiv:2006.01038.
14. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
15. He, K.; Zhang, X.; Ren, S.; Sun, J. Identity mappings in deep residual networks. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; pp. 630–645.

16. Chen, J.; Lopresti, D. Table detection in noisy off-line handwritten documents. In Proceedings of the 2011 International Conference on Document Analysis and Recognition, Beijing, China, 18–21 September 2011; pp. 399–403.
17. Fang, J.; Gao, L.; Bai, K.; Qiu, R.; Tao, X.; Tang, Z. A table detection method for multipage pdf documents via visual separators and tabular structures. In Proceedings of the 2011 International Conference on Document Analysis and Recognition, Beijing, China, 18–21 September 2011; pp. 779–783.
18. Shafait, F.; Smith, R. Table detection in heterogeneous documents. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, Boston, MA, USA, 9–11 June 2010; pp. 65–72.
19. Tupaj, S.; Shi, Z.; Chang, C.H.; Alam, H. *Extracting Tabular Information from Text Files*; EECS Department, Tufts University: Medford, OR, USA, 1996.
20. Wangt, Y.; Phillipst, I.T.; Haralick, R. Automatic table ground truth generation and a background-analysis-based table structure extraction method. In Proceedings of the Sixth International Conference on Document Analysis and Recognition, Seattle, WA, USA, 10–13 September 2001; pp. 528–532.
21. Kieninger, T.; Dengel, A. A paper-to-HTML table converting system. In Proceedings of the Document Analysis Systems (DAS), Nagano, Japan, 4–6 November 1998; Volume 98, pp. 356–365.
22. Kieninger, T.; Dengel, A. Table recognition and labeling using intrinsic layout features. In *International Conference on Advances in Pattern Recognition*; Springer: London, UK, 1999; pp. 307–316.
23. Kieninger, T.; Dengel, A. Applying the T-RECS table recognition system to the business letter domain. In Proceedings of the Sixth International Conference on Document Analysis and Recognition, Seattle, WA, USA, 10–13 September 2001; pp. 518–522.
24. Gatos, B.; Danatsas, D.; Pratikakis, I.; Perantonis, S.J. Automatic table detection in document images. In *International Conference on Pattern Recognition and Image Analysis*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 609–618.
25. e Silva, A.C. Learning rich hidden markov models in document analysis: Table location. In Proceedings of the 2009 10th International Conference on Document Analysis and Recognition, Barcelona, Spain, 26–29 July 2009; pp. 843–847.
26. Hu, J.; Kashi, R.S.; Lopresti, D.P.; Wilfong, G. Medium-independent table detection. In *Document Recognition and Retrieval VII*; International Society for Optics and Photonics: Bellingham, WA, USA, 1999; Volume 3967, pp. 291–302.
27. Siddiqui, S.A.; Malik, M.I.; Agne, S.; Dengel, A.; Ahmed, S. Decnt: Deep deformable cnn for table detection. *IEEE Access* **2018**, *6*, 74151–74161. [[CrossRef](#)]
28. Younas, J.; Siddiqui, S.A.; Munir, M.; Malik, M.I.; Shafait, F.; Lukowicz, P.; Ahmed, S. Fi-Fo Detector: Figure and Formula Detection Using Deformable Networks. *Appl. Sci.* **2020**, *10*, 6460. [[CrossRef](#)]
29. Schreiber, S.; Agne, S.; Wolf, I.; Dengel, A.; Ahmed, S. Deepdesrt: Deep learning for detection and structure recognition of tables in document images. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 1162–1167.
30. Vo, N.D.; Nguyen, K.; Nguyen, T.V.; Nguyen, K. Ensemble of deep object detectors for page object detection. In Proceedings of the 12th International Conference on Ubiquitous Information Management and Communication, Langkawi, Malaysia, 5–7 January 2018; pp. 1–6.
31. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448.
32. Saha, R.; Mondal, A.; Jawahar, C.V. Graphical object detection in document images. In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 51–58.
33. Gilani, A.; Qasim, S.R.; Malik, I.; Shafait, F. Table detection using deep learning. In Proceedings of the 2017 14th IAPR international conference on document analysis and recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 771–776.
34. Agarwal, M.; Mondal, A.; Jawahar, C.V. CDeC-Net: Composite Deformable Cascade Network for Table Detection in Document Images. *arXiv* **2020**, arXiv:2008.10831.
35. Kavasidis, I.; Palazzo, S.; Spampinato, C.; Pino, C.; Giordano, D.; Giuffrida, D.; Messina, P. A saliency-based convolutional neural network for table and chart detection in digitized documents. In Proceedings of the International Conference on Image Analysis and Processing, Trento, Italy, 9–13 September 2019; pp. 292–302.
36. Yi, X.; Gao, L.; Liao, Y.; Zhang, X.; Liu, R.; Jiang, Z. CNN based page object detection in document images. In Proceedings of the 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), Kyoto, Japan, 9–15 November 2017; Volume 1, pp. 230–235.
37. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2014; pp. 818–833.
38. Göbel, M.; Hassan, T.; Oro, E.; Orsi, G. ICDAR 2013 table competition. In Proceedings of the 2013 12th International Conference on Document Analysis and Recognition, Washington, DC, USA, 25–28 August 2013; pp. 1449–1453.
39. Everingham, M.; Van Gool, L.; Williams, C.K.; Winn, J.; Zisserman, A. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [[CrossRef](#)]
40. Younas, J.; Rizvi, S.T.R.; Malik, M.I.; Shafait, F.; Lukowicz, P.; Ahmed, S. FFD: Figure and formula detection from document images. In Proceedings of the 2019 Digital Image Computing: Techniques and Applications (DICTA), Perth, Australia, 2–4 December 2019; pp. 1–7.
41. Breu, H.; Gil, J.; Kirkpatrick, D.; Werman, M. Linear time Euclidean distance transform algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.* **1995**, *17*, 529–533. [[CrossRef](#)]

42. Fabbri, R.; Costa, L.D.F.; Torelli, J.C.; Bruno, O.M. 2D Euclidean distance transform algorithms: A comparative survey. *ACM Comput. Surv. (CSUR)* **2008**, *40*, 1–44. [[CrossRef](#)]
43. Ragnemalm, I. The Euclidean distance transform in arbitrary dimensions. *Pattern Recognit. Lett.* **1993**, *14*, 883–888. [[CrossRef](#)]
44. Shahab, A.; Shafait, F.; Kieninger, T.; Dengel, A. An open approach towards the benchmarking of table structure recognition systems. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, Boston, MA, USA, 26–29 July 2010; pp. 113–120.
45. Fang, J.; Tao, X.; Tang, Z.; Qiu, R.; Liu, Y. Dataset, ground-truth and performance metrics for table detection evaluation. In Proceedings of the 2012 10th IAPR International Workshop on Document Analysis Systems, Gold Coast, Australia, 27–29 March 2012; pp. 445–449.
46. Li, M.; Cui, L.; Huang, S.; Wei, F.; Zhou, M.; Li, Z. Tablebank: Table benchmark for image-based table detection and recognition. In Proceedings of the 12th Language Resources and Evaluation Conference, Marseille, France, 11–16 May 2020; pp. 1918–1925.
47. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable convolutional networks. In Proceedings of the IEEE international Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 764–773.
48. Neubeck, A.; Van Gool, L. Efficient non-maximum suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; Volume 3, pp. 850–855.
49. Krähenbühl, P.; Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. *Adv. Neural Inf. Process. Syst.* **2011**, *24*, 109–117.
50. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
51. Mondal, A.; Lipps, P.; Jawahar, C.V. IIT-AR-13K: A new dataset for graphical object detection in documents. In Proceedings of the International Workshop on Document Analysis Systems, Wuhan, China, 17–20 May 2020; pp. 216–230.
52. Gao, L.; Huang, Y.; Déjean, H.; Meunier, J.L.; Yan, Q.; Fang, Y.; Lang, E. ICDAR 2019 competition on table detection and recognition (cTDaR). In Proceedings of the 2019 International Conference on Document Analysis and Recognition (ICDAR), Sydney, Australia, 20–25 September 2019; pp. 1510–1515.
53. Siegel, N.; Lourie, N.; Power, R.; Ammar, W. Extracting scientific figures with distantly supervised neural networks. In Proceedings of the 18th ACM/IEEE on Joint Conference on Digital Libraries, Fort Worth, TX, USA, 3–7 June 2018; pp. 223–232.
54. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C.L. Microsoft coco: Common objects in context. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 740–755.
55. Hashmi, K.A.; Liwicki, M.; Stricker, D.; Afzal, M.A.; Afzal, M.A.; Afzal, M.Z. Current Status and Performance Analysis of Table Recognition in Document Images with Deep Neural Networks. *arXiv* **2021**, arXiv:2104.14272.
56. Chen, K.; Pang, J.; Wang, J.; Xiong, Y.; Li, X.; Sun, S.; Lin, D. Hybrid task cascade for instance segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, FL, USA, 15–21 June 2019; pp. 4974–4983.
57. Cai, Z.; Vasconcelos, N. Cascade r-cnn: Delving into high quality object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 6154–6162.
58. Phong, B.H.; Hoang, T.M.; Le, T.L. A hybrid method for mathematical expression detection in scientific document images. *IEEE Access* **2020**, *8*, 83663–83684. [[CrossRef](#)]
59. Liu, Y.; Wang, Y.; Wang, S.; Liang, T.; Zhao, Q.; Tang, Z.; Ling, H. Cbnet: A novel composite backbone network architecture for object detection. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34, pp. 11653–11660.
60. Qiao, S.; Chen, L.C.; Yuille, A. Detectors: Detecting objects with recursive feature pyramid and switchable atrous convolution. *arXiv* **2020**, arXiv:2006.02334.
61. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv* **2021**, arXiv:2103.14030.
62. Gorai, M.; Nene, M.J. Layout and Text Extraction from Document Images using Neural Networks. In Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 10–12 June 2020; pp. 1107–1112.
63. Xu, Y.; Li, M.; Cui, L.; Huang, S.; Wei, F.; Zhou, M. Layoutlm: Pre-training of text and layout for document image understanding. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual Event, CA, USA, 6–10 July 2020; pp. 1192–1200.
64. Jiao, L.; Zhang, F.; Liu, F.; Yang, S.; Li, L.; Feng, Z.; Qu, R. A survey of deep learning-based object detection. *IEEE Access* **2019**, *7*, 128837–128868. [[CrossRef](#)]
65. Liu, Y.; Sun, P.; Wergeles, N.; Shang, Y. A survey and performance evaluation of deep learning methods for small object detection. *Expert Syst. Appl.* **2021**, *114602*, 172.
66. Chen, Y.; Yang, T.; Zhang, X.; Meng, G.; Pan, C.; Sun, J. Detnas: Neural architecture search on object detection. *arXiv* **2019**, arXiv:1903.10979.
67. Wang, N.; Gao, Y.; Chen, H.; Wang, P.; Tian, Z.; Shen, C.; Zhang, Y. NAS-FCOS: Fast neural architecture search for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–18 June 2020; pp. 11943–11951.

68. Elsken, T.; Metzen, J.H.; Hutter, F. Neural architecture search: A survey. *J. Mach. Learn. Res.* **2019**, *20*, 1–21.
69. Lindauer, M.; Hutter, F. Best practices for scientific research on neural architecture search. *J. Mach. Learn. Res.* **2020**, *21*, 1–18.