A New Annotation Method and Dataset for Layout Analysis of Long Documents

Aman Ahuja
Virginia Tech
Blacksburg, VA, USA
aahuja@vt.edu

Kevin Dinh
Virginia Tech
Blacksburg, VA, USA
kevinqdinh@vt.edu

Brian Dinh
Virginia Tech
Blacksburg, VA, USA
briandinh01@vt.edu

William A. Ingram
Virginia Tech
Blacksburg, Virginia, USA
waingram@vt.edu

Edward A. Fox
Virginia Tech
Blacksburg, VA, USA
fox@vt.edu

ABSTRACT

Parsing long documents, such as books, theses, and dissertations, is an important component of information extraction from scholarly documents. Layout analysis methods based on object detection have been developed in recent years to help with PDF document parsing. However, several challenges hinder the adoption of such methods for scholarly documents such as theses and dissertations. These include (a) the manual effort and resources required to annotate training datasets, (b) the scanned nature of many documents and the inherent noise present resulting from the capture process, and (c) the imbalanced distribution of various types of elements in the documents. In this paper, we address some of the challenges related to object detection based layout analysis for scholarly long documents. First, we propose an AI-aided annotation method to help develop training datasets for object detection based layout analysis. This leverages the knowledge of existing trained models to help human annotators, thus reducing the time required for annotation. It also addresses the class imbalance problem, guiding annotators to focus on labeling instances of rare classes. We also introduce ETD-ODv2, a novel dataset for object detection on electronic theses and dissertations (ETDs). In addition to the page images included in ETD-OD [1], our dataset consists of more than 16K manually annotated page images originating from 100 scanned ETDs, along with annotations for 20K page images primarily consisting of rare classes that were labeled using the proposed framework. The new dataset thus covers a diversity of document types, viz., scanned and born-digital, and is better balanced in terms of training samples from different object categories.

CCS CONCEPTS

• **Applied computing** → **Document analysis**; **Annotation**; *Extensible Markup Language (XML)*; • **Computing methodologies** → **Object detection**; • **Information systems** → **Digital libraries and archives**.

WWW '23 Companion, April 30–May 04, 2023, Austin, TX, USA

ACM ISBN 978-1-4503-9419-2/23/04.
https://doi.org/10.1145/3543873.3587609

KEYWORDS

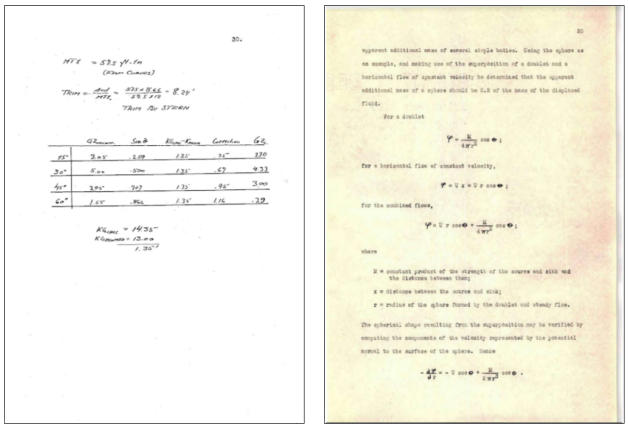
Object Detection, Scholarly Documents, Electronic Theses and Dissertations, Document Understanding, AI-Aided

ACM Reference Format:
Aman Ahuja, Kevin Dinh, Brian Dinh, William A. Ingram, and Edward A. Fox. 2023. A New Annotation Method and Dataset for Layout Analysis of Long Documents. In *Companion Proceedings of the ACM Web Conference 2023 (WWW '23 Companion), April 30–May 04, 2023, Austin, TX, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3543873.3587609

1 INTRODUCTION

Long scholarly documents, such as e-books and electronic theses and dissertations (ETDs), contain valuable knowledge that is of interest to many in the academic and research community. These documents often exist on the web in PDF form, making it difficult to extract information from such documents to support end-user services. To make them compatible with modern digital library services, such as document search and browsing, the information from these documents first needs to be extracted and converted into a machine-friendly format such as XML. This is also essential for improving the accessibility of such documents, as accessibility tools such as on-screen readers often require that different elements of a document be tagged. Owing to the wide range of variation that is observed in such documents, as a result of specific layouts used by different institutions, as well as the variation in writing style and elements used across different scientific domains, parsing scholarly PDF documents is a nontrivial task.

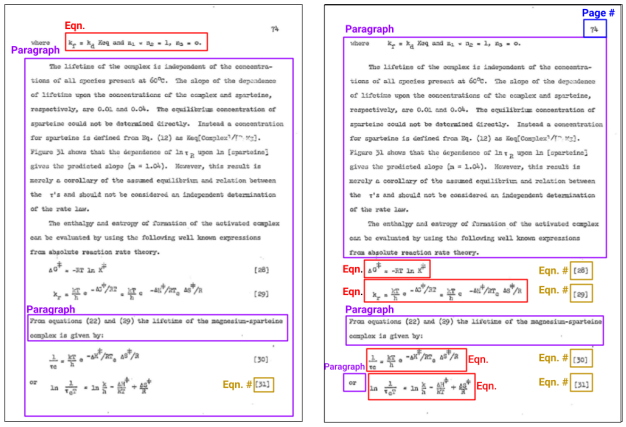
In recent years, with advances in the field of deep learning, several techniques have been proposed to help with the problem of information extraction from PDF documents. One research direction that has shown promising results for the problem of layout analysis and parsing of PDF documents is based on object detection. Object detection [4, 13] aims to identify and extract objects of interest from an input image, using bounding boxes and associated labels for each element in the image. The resultant elements, such as figures, tables, and paragraphs, can then be used for further downstream tasks. Object detection-based layout analysis has recently been studied for scholarly documents such as research papers [10, 21] and ETDs [1]. These investigations mainly focus on digital PDF documents that are typically prepared using text editing software such as LaTeX or Microsoft Word.



(a) Handwritten elements. (b) Noisy patches.

Figure 1: Examples of pages from scanned documents.

An important aspect of object detection-based methods is that they often require a huge amount of labeled training data. For digital documents, especially those written in LaTeX, it is often possible to obtain annotations using rule-based automatic annotation methods [10]. However, in the case of scanned documents, as well as digital documents without accompanying LaTeX source code, annotating data is a cumbersome process that requires a great amount of manual effort. In the case of ETDs, many documents present in digital libraries, especially the older ones, tend to be scanned documents that were written using legacy text editing software or with a typewriter. These documents were then microfilmed and/or scanned and converted to PDF. Consequently, these documents contain a large amount of noise that was introduced during the PDF conversion process, as shown in Figure 1. Furthermore, given



(a) Model generated annotations. (b) Corrected annotations.

Figure 2: An illustration showing a page from a scanned document, the annotations generated by an object detection model trained on a small dataset, and the final annotations after correction by a human annotator.

that these documents were prepared using legacy methods, they differ significantly from newer documents, such as digital ETDs, in terms of layout and structure. Additionally, some of the elements, such as metadata elements like ETD title and author name, can only be found on a few pages, while others, such as a paragraph, can be found on many pages in a document. As such, the distribution of different object categories in the training data varies. This also affects the performance of object detection models in classes with a limited number of training instances.

In this paper, we propose an AI-aided annotation framework to minimize the amount of resources such as annotation time associated with developing training datasets for layout analysis. Our proposed framework utilizes the predictive capabilities of models trained on existing datasets to assist human annotators. As illustrated in Figure 2, although the annotations generated by the model might not be completely correct, many of them are often correct, which can reduce the number of instances that need to be manually labeled. This significantly speeds up the annotation process, without compromising the quality of the generated dataset. It also helps to address the problem of class imbalance in object detection datasets, by guiding annotators to label images that are more likely to contain elements from a predefined set. Experimental results show that our proposed annotation scheme significantly reduces the annotation time and class imbalance, thus resulting in models with improved performance on various object classes. We also introduce ETD-ODv2, a new dataset for object detection-based layout analysis of long documents such as theses and dissertations. ETD-ODv2 supplements the page images included in ETD-OD, adding 20K page images originating from scanned theses and dissertations. It also adds annotations for page images that are likely to contain low-frequency elements, such as *document title* and *algorithm*, since they can only be found on selected pages of a document or documents from specific domains. These pages were sourced from a large corpus consisting of both scanned and digital documents, making them helpful for mitigating the class imbalance in existing datasets as well. It thus addresses the limitations of existing datasets for ETD layout analysis, whose scope is limited to digital documents only, and suffers from a class imbalance problem. Our experimental results show that models trained on our newly annotated dataset perform much better than those trained on other datasets.

The contributions of this work are threefold.

- We propose an AI-aided annotation scheme to develop training datasets for layout analysis. The proposed annotation scheme significantly reduces the annotation time, while also allowing us to address the class imbalance problem in training datasets.
- We introduce **ETD-ODv2**, a new manually annotated dataset for object detection-based layout analysis of scanned theses and dissertations. Unlike existing datasets that focus mainly on digital documents, our dataset helps with the layout analysis of scanned documents.
- We show that training on datasets focused towards certain low-frequency elements significantly improves the performance of layout analysis methods. Our dataset also consists of labeled images of pages that are likely to contain elements from rare classes to help alleviate the performance issues that arise as a result of class imbalance in object detection datasets.

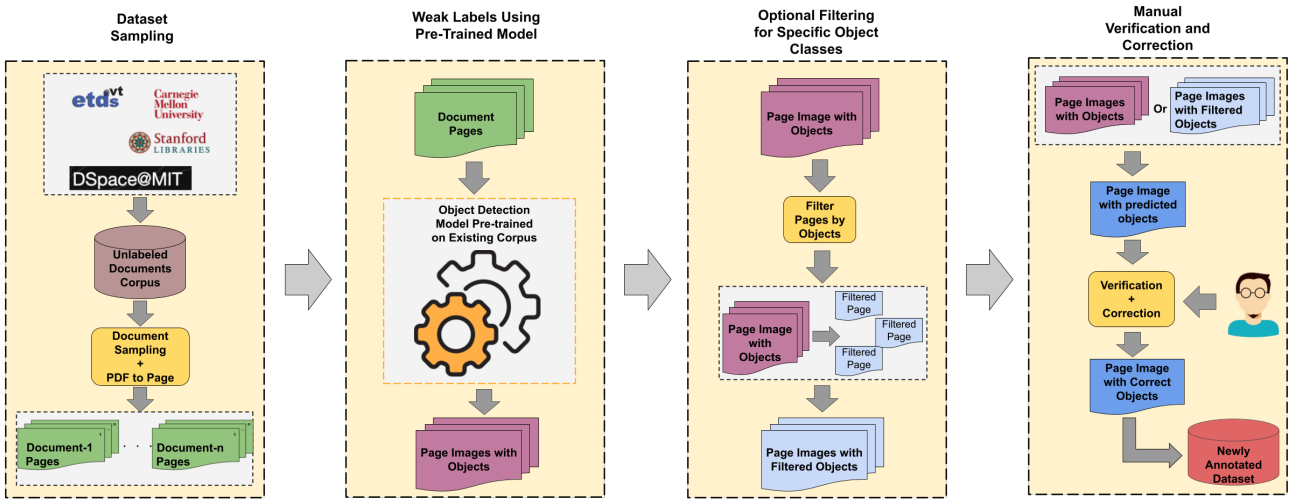


Figure 3: Architecture of the proposed AI-aided annotation framework.

2 RELATED WORK

Information extraction from documents has gained popularity in recent years due to its value for various types of scholarly literature, as well as business documents such as insurance claims, tax forms, and resumes. We review some of the related datasets, techniques, and annotation methods in the document layout analysis domain.

2.1 Layout Analysis Datasets

Several datasets have recently been developed to help with the task of extracting information from documents. These datasets cover a diverse range of document types. In the case of scholarly documents, two types of documents are studied: short documents, such as research papers, and long documents, such as theses and dissertations. TableBank [9] consists mainly of annotations for a specific object type, such as tables. Datasets such as PubLayNet [21] and DocBank [10] cover a wider range of object types to support more in-depth layout analysis. All of these datasets are based primarily on research papers from open-access repositories. Layout analysis for long scholarly documents, such as books, theses, and dissertations, has also attracted interest from the research community. Some works along this line include ScanBank [8], which supports figure extraction, and ETD-OD [1], which supports the extraction of a diverse set of element types found in theses and dissertations.

Due to its application in other business domains such as insurance, tax, and healthcare, information extraction from form-like documents has become a popular research area. Datasets in this category include FUNSD [7] and CORD [12], which are based on forms and receipts, respectively. SROIE [6] is another dataset based on receipts, while KLEISTER [15] is based on longer documents such as non-disclosure agreements.

2.2 Layout Analysis Methods

Traditional methods for document layout analysis included rule-based methods [2] and text-based methods (semantics) [11]. With the advent of deep learning-based methods for object detection such as Fast-RCNN [4], Faster-RCNN [13], and YOLO [17], document layout analysis based on visual features gained popularity [1, 14]. More recently, techniques that use visual and semantic features for pre-training, such as LayoutLM (v1 [19], v2 [20], v3 [5]), have been proposed. The pre-trained models can then be fine-tuned on downstream tasks like object detection.

2.3 Annotation Methods

Due to the intensive nature of dataset annotation in terms of time and cost, researchers have proposed several techniques to annotate training datasets for object detection models. For PDF documents with an accompanying MS-Word, XML, or LaTeX file, automatic extraction based on tags is possible [9, 10]. However, in the case of scanned documents, a rule-based approach cannot be used. In such cases, techniques have been explored that can help annotators or guide them in annotating samples about which the model is most uncertain [22].

3 PROPOSED AI-AIDED ANNOTATION SCHEME

Due to the resource-intensive nature of the dataset annotation process, labeled data for training supervised machine learning models are always scarce. However, unlabeled data are generally available in abundance. This is also the case with document layout analysis, where getting high-quality annotations for documents and their respective pages is not easy. However, given the numerous documents that exist on the Internet and in digital libraries, many unlabeled scholarly documents are publicly available. Although labeling document page images is a cumbersome task, we hypothesize that models trained on existing datasets can be used to assist human annotators in the labeling process, thus reducing the time required to annotate training datasets. These models can be used to generate weak labels for the huge corpus of unlabeled ETDs, which

can then be filtered, validated, and corrected by human annotators. Based on this assumption, in this section, we propose an AI-aided annotation framework for developing datasets to train supervised object detection models. Figure 3 gives an overview of our proposed framework. The key components of this framework are discussed in detail below.

3.1 Dataset Sampling

We use a large corpus of unlabeled ETDs, sourced from multiple open access digital libraries. We first sample a set of documents from this unlabeled corpus that can be used for AI-aided annotation. Each of these documents is then split into page images, since object detection models require images as input.

3.2 Weak Labels Using Pre-Trained Model

Once we have a set of documents as well as their respective page images, they are sent to an object detection model such as YOLO [17] or Faster-RCNN [13] that has been pre-trained on an existing labeled dataset, such as ETD-OD [1]. The labels thus inferred for each image serve as weak annotations for further processing and manual verification/annotation.

3.3 Optional Filtering for Specific Object Classes

In some cases, such as in the case of academic documents like theses and dissertations, labeling the entire set of pages found in the sampled documents could result in a highly unbalanced dataset. In such cases, it might be desirable to use weak labels to filter out images containing a pre-defined set of object categories. We refer to these object categories as objects of interest. These categories include minority classes, such as those containing very few instances in the labeled dataset, or those that have lower performance as compared to other categories. This could enable researchers to produce datasets with balanced class distributions.

3.4 Manual Verification and Correction

The filtered set of pages, along with their predicted bounding boxes and their respective labels, is then verified by human annotators for correctness. For page images with correctly predicted objects, no changes are made and the respective page is added to the verified dataset. For page images with incorrect predictions, whether in terms of missing or incorrect labels, the correct bounding boxes are drawn by human annotators before being added to the verified dataset.

The new dataset can then be used to fine-tune existing pre-trained models or in combination with existing datasets for model training.

4 ETD-ODV2 DATASET

In this section, we introduce ETD-ODv2, a new dataset for layout analysis of electronic theses and dissertations. Although existing datasets like ETD-OD [1] can be helpful in layout extraction from digital documents, they suffer from a class imbalance problem and do not contain scanned documents.

4.1 Scanned Documents

There are several attributes related to scanned documents that are not found in digital documents. These include the following.

- **Noisy patches:** A common observation found in scanned documents is that a large number of pages contain noisy patches that result from the process of converting such documents into an electronically readable PDF file.
- **Low resolution:** Given that these documents are essentially images of hard-copy versions of the original document, they tend to have relatively low resolution.
- **Dilated or eroded text:** Another common observation regarding many scanned documents is that the text is eroded (i.e., has a thinner font than the original document) or dilated. This can also be attributed to the PDF conversion process.
- **Handwritten elements:** Some of the pages of scanned documents contain elements – such as tables, figures, and equations – that were written or drawn by hand and were not typed or created using software.

Due to the presence of such attributes, object detection models trained on the digital documents dataset generally do not perform well on scanned documents. Hence, our new dataset includes manually annotated page images from scanned documents, to support layout analysis on scanned documents.

4.2 Page Images with Minority Elements

While it is desirable to have images of pages from scanned documents, this does not prevent the dataset from being subject to a class imbalance problem. This is because some elements – such as *document title* and *author name* – typically only appear on a small set of pages in the document, such as the front page. Therefore, a dataset constructed by labeling all pages appearing in a document will always be prone to the class imbalance problem. Moreover, some element classes such as *algorithm* might only appear in documents in certain domains, such as computer science. Hence, a set of documents uniformly sampled from several different domains will have few pages with such instances. To alleviate this problem, we use the proposed AI-aided annotation method to filter and annotate pages that are more likely to contain such minority elements. These page images were sourced from both digital and scanned documents. The elements that we consider to be minority elements are listed below.

- **Elements found on a limited number of pages:** Title, Author, Date, University, Committee, Degree, Abstract Text, List of Contents Heading.
- **Elements found in documents from select disciplines:** Equation, Equation Number, Algorithm, Reference Heading.

4.3 Dataset Source and Object Classes

To ensure compatibility with existing datasets, we use the object categories defined in ETD-OD for annotation. The documents in both subsets of our data set (i.e., the scanned and AI-aided) were sourced from a uniformly sampled set of theses and dissertations from open access institutional repositories of US origin [16].

Category Name	Description	#Digital Instances	#Scanned Instances	#AI-Aided Instances	#Total Instances
Title	Title of the document	439 (0.4%)	253 (0.4%)	2186 (1.6%)	2878 (1.0%)
Author	Name of the document author	404 (0.4%)	249 (0.4%)	2548 (1.9%)	3201 (1.1%)
Date	Date of publication, or of final research defense	324 (0.3%)	224 (0.4%)	2415 (1.8%)	2963 (1.0%)
University	University/institution of the author	340 (0.3%)	203 (0.3%)	1873 (1.4%)	2416 (0.8%)
Committee	Committee that approved the document	305 (0.3%)	83 (0.1%)	1472 (1.1%)	1860 (0.6%)
Degree	Degree (e.g., Master of Science) being earned.	281 (0.3%)	202 (0.3%)	1834 (1.3%)	2317 (0.8%)
Abstract Heading	A header that indicates the start of abstract text	169 (0.2%)	113 (0.2%)	807 (0.6%)	1089 (0.4%)
Abstract Text	The actual text of the abstract	183 (0.2%)	73 (0.1%)	952 (0.7%)	1208 (0.4%)
List of Contents Heading	A header that identifies the content of a list	512 (0.5%)	300 (0.5%)	3151 (2.3%)	3963 (1.3%)
List of Contents Text	The actual list of entries for the type of content	1059 (1.1%)	460 (0.7%)	3172 (2.3%)	4691 (1.6%)
Chapter Title	The title of the chapter	2199 (2.2%)	1926 (3.1%)	1263 (0.9%)	5388 (1.8%)
Section	The header of a section which splits a document	9337 (9.4%)	2946 (4.7%)	5196 (3.8%)	17479 (5.8%)
Paragraph	The main textual content of the document	30359 (30.4%)	17962 (28.5%)	34601 (25.2%)	82922 (27.6%)
Figure	A figure, chart, or other visual illustration	6359 (6.4%)	2977 (4.7%)	2148 (1.6%)	11484 (3.8%)
Figure Caption	The text caption that describes a figure	5722 (5.7%)	2370 (3.8%)	1564 (1.1%)	9656 (3.2%)
Table	The table element category	3145 (3.1%)	2192 (3.5%)	656 (0.5%)	5993 (2.0%)
Table Caption	The text caption that describes a table	2225 (2.2%)	1872 (3.0%)	399 (0.3%)	4496 (1.5%)
Equation	A mathematical equation/formula	5092 (5.1%)	5579 (8.8%)	27266 (19.8%)	37937 (12.6%)
Equation Number	Used to reference an equation with a number	1834 (1.8%)	3727 (5.9%)	20943 (15.2%)	26504 (8.8%)
Algorithm	An algorithm description, e.g., as pseudo-code	96 (0.1%)	224 (0.4%)	787 (0.6%)	1107 (0.4%)
Footnote	Auxiliary information at the end of content	2029 (2.0%)	2340 (3.7%)	1045 (0.8%)	5414 (1.8%)
Page Number	A number of a specific page in a document	24543 (24.6%)	15800 (25.0%)	17454 (12.7%)	57797 (19.2%)
Reference Heading	A header that indicates the start of a reference list	271 (0.3%)	189 (0.3%)	1830 (1.3%)	2290 (0.8%)
Reference Text	The actual list of reference cited in the document	2632 (2.6%)	864 (1.4%)	1839 (1.3%)	5335 (1.8%)
Total Objects		99859	63128	137401	300388
Total Images		25073	16766	20204	62043

Table 1: ETD-ODv2 dataset statistics.

4.4 Dataset Statistics

Table 1 shows the detailed statistics of different object categories in our dataset.

4.4.1 Scanned Documents: The subset of scanned documents in our dataset consists of images and bounding box annotations of ~16K pages, derived from 100 theses and dissertations. These documents were annotated by a group of five undergraduate students [23]. To ensure the correctness, each sample also went through another round of review by one of the authors. We use Roboflow[1] as the dataset annotation platform.

4.4.2 Pages with low-frequency elements: Our dataset also consists of ~20K page images from ~1,200 documents that were annotated using our proposed AI-aided annotation framework. The pages were then filtered based on the labels listed above and reviewed and corrected as needed by a group of four annotators [3].

5 EXPERIMENTS

In this section, we report the experimental results obtained during our evaluation. Our experiments focus on determining the improvements in terms of human resources, such as annotation time, obtained using the AI-aided annotation strategy. We also analyze whether the new dataset consisting of scanned documents and pages with instances from lower-frequency categories can be helpful in improving the performance of object detection models.

5.1 Annotation Time

5.1.1 Experimental Setup: To construct our proposed AI-aided annotation framework, we used the bounding box widget from the open source framework `pylabel`[2], which was integrated with a pre-trained object detection model. We trained a YOLOv7 model [17] on ETD-OD [1] and a small set of ~2K scanned documents. We only used a small number of samples from the scanned documents dataset, as that was the only sample available at the time. The model obtained was then used in our AI-aided framework to generate the proposed labels. We will refer to this model as **YOLOv7_base** in the remainder of the discussion. As noted in [1], YOLOv7 outperforms other models in the object detection task, so we use it as the detection model for empirical evaluation.

5.1.2 Evaluation Settings: To determine whether the proposed AI-aided annotation scheme reduces resource requirements, we compare the time required to label images under different settings.

[1] https://roboflow.com/

[2] https://pylabel.readthedocs.io/en/latest/

- **No Model Assistance:** This is the classical labeling setting under which the annotators are shown neither bounding boxes nor the respective labels for page images.
- **AI-Aided-v1:** Under this setting, for each image, the annotators were shown the bounding boxes generated by the **YOLOv7_base** model.
- **AI-Aided-v2:** For this setting, we fine-tuned the **YOLOv7_base** model on a set of 10K page images labeled using our AI-aided annotation scheme. This was done to evaluate whether the assistance of a model trained on an additional new dataset affects the annotation time. We then used this model to generate bounding boxes for each image shown to the annotators.

In the two AI-Aided settings, annotators were asked first to review the model-generated annotations. All correct annotations were left unchanged, and only missing, incorrect, or extra-bounding boxes were asked to be modified. For each of the three settings, each of the four annotators annotated ~500 pages, and the time spent on annotation was recorded.

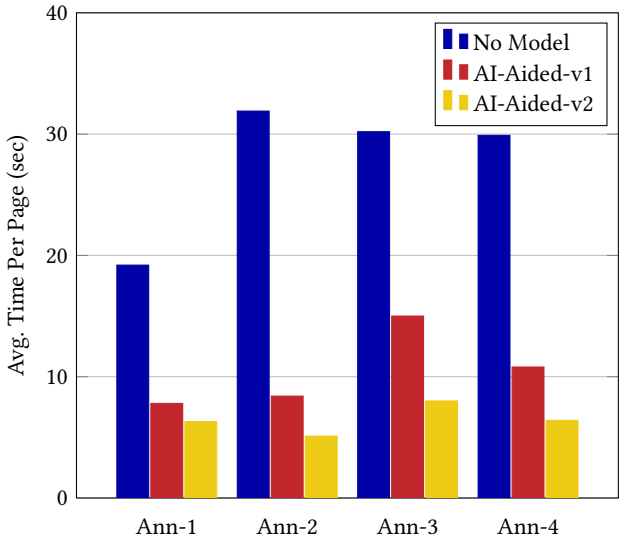


Figure 4: Annotation time for each annotator under different annotation settings.

5.1.3 Results: In Figure 4, we report the average time spent per page by each of the annotators under different annotation settings. The following observations can be made:

- **Model assistance significantly reduces annotation time:** As we can observe from the graph, the average time required to annotate a page without the assistance of a model (i.e., without any proposed bounding boxes) is 2-3 times longer than for each of the AI-aided settings. This is likely because even though the models used for assisting annotators might have been trained on limited data and coverage (in terms of document types and object classes), they still possess predictive power to help with many of the elements found in pages, such as paragraphs and figures. Thus, we can conclude that the assistance of models trained on existing data significantly helps in annotating more data by reducing the time required for annotation.
- **Model assistance increases with better trained models:** Another observation that can be made from Figure 4 is that as we obtain models with better predictive power, the suggested labels of the model become more accurate, further reducing the time required to annotate a page. The model used for the `AI-Aided-v2` setting had been trained on 10K more samples than the one used in `AI-Aided-v1` setting. The samples used were also more balanced in terms of object classes. Therefore, it has better predictive power, enabling it to be more helpful to human annotators in annotating more data.

5.2 Object Detection Performance

In this analysis, we present our findings on how the AI-aided annotated dataset helps improve object detection performance. The specific details of this analysis are described below.

5.2.1 Object Detection Model: As stated above, we use YOLOv7 as the benchmark object detection model for this analysis. Since the purpose of this analysis is to determine how training on different datasets impacts model performance, the specific choice of object detection model is beyond the scope of this analysis. Moreover, previous studies have shown that YOLOv7 is the state-of-the-art model for object detection tasks [1, 17, 18].

5.2.2 Test Dataset: Since the AI-aided subset of our dataset was constructed with the objective of mitigating the class imbalance problem, it consists of page images from documents of several types, such as scanned and digital. Therefore, to analyze how training with the AI-aided dataset helps object detection models on various types of documents, we construct a test dataset consisting of page images sampled from ETD-OD [1], as well as the scanned and low-frequency element pages from ETD-ODv2. This is done to ensure that the test set is representative of diversity in terms of both document types and object types. The breakdown of images and objects in the test dataset is shown in Table 2.

Source	# Images	# Objects
Digital	3760	14319
Scanned	9353	9294
AI-Aided	3031	20718
Total Test	9353	44331

Table 2: Distribution of the test dataset.

5.2.3 Baselines: We use the versions of the dataset listed below to evaluate object detection performance. All versions used YOLOv7 as the object detection model. The number of images and objects in each version is listed in Table 3.

- **Digital:** This version of the model was trained only on the digital document images from ETD-OD. As such, the training dataset contained a small number of samples from the minority classes due to the class imbalance in the scanned subset.

- **Scanned:** This version of the model was trained only on the scanned subset of the ETD-ODv2 dataset. As in the previous setting, the training dataset used in this setting also has the class imbalance problem.
- **Digital + Scanned:** Under this setting, the YOLOv7 model was trained on the combined images of scanned and digital documents, that is, a merged set consisting of the two dataset splits described above.
- **Digital + Scanned + AI-Aided:** This setting uses the **Digital + Scanned** split described above, along with the AI-aided subset of ETD-ODv2. This setting represents a model that has been trained on diverse types of document (i.e., digital and scanned) and consists of a larger number of training instances from each object category.

Version	# Images	# Objects
Digital	21313	85540
Scanned	14204	53834
Digital + Scanned	35517	139374
Digital + Scanned + AI-Aided	52690	256057

Table 3: Statistics of different versions of the data set used for training.

5.2.4 Evaluation Metrics: We use the two commonly used object detection metrics to evaluate the results of different models discussed above. Both metrics are based on the average precision (AP), which is calculated based on the number of predicted objects that overlap with the ground-truth object over a certain threshold in terms of the area. The two metrics are described in detail below.

- **AP@0.50 / mAP@0.50:** For a given object category, AP@0.50 is the percentage of predicted bounding boxes that overlap with the true bounding boxes by more than 50% in terms of area. mAP@0.50 is the average of AP@0.50 for all object categories.
- **AP@0.50:0.95 / mAP@0.50:0.95:** This is calculated by first calculating the AP at different thresholds, from 0.50 to 0.95, with a step of 0.05. All these AP values are averaged to compute AP@0.50:0.95 for an object category. mAP@0.50:0.95 is the average of AP@0.50:0.95 for all object categories.

5.2.5 Results: Table 4 shows the results obtained on the test dataset described above in each of the training settings. Based on the results shown, the following observations can be made:

- **Performance w.r.t. document type:** The subset of images used to train the **Scanned** model had the highest amount of noise and

	AP@0.5				AP@0.5:0.95			
Categories	**Digital**	**Scanned**	**Digital+ Scanned**	**Digital+ Scanned+ AI-Aided**	**Digital**	**Scanned**	**Digital+ Scanned**	**Digital+ Scanned+ AI-Aided**
Title	0.861	0.538	0.888	0.924	0.688	0.340	0.672	0.732
Author	0.814	0.471	0.833	0.927	0.556	0.221	0.523	0.624
Date	0.676	0.393	0.731	0.852	0.454	0.124	0.398	0.545
University	0.730	0.312	0.788	0.874	0.539	0.156	0.529	0.628
Committee	0.822	0.327	0.856	0.926	0.622	0.167	0.620	0.692
Degree	0.524	0.060	0.551	0.732	0.385	0.024	0.380	0.532
Abstract Heading	0.897	0.320	0.929	0.948	0.636	0.127	0.628	0.672
Abstract Text	0.812	0.703	0.837	0.872	0.786	0.629	0.811	0.845
List of Contents Heading	0.880	0.782	0.884	0.915	0.655	0.293	0.555	0.690
List of Contents Text	0.939	0.926	0.955	0.966	0.875	0.790	0.889	0.896
Chapter Title	0.503	0.460	0.761	0.786	0.273	0.211	0.406	0.425
Section	0.861	0.706	0.882	0.890	0.495	0.306	0.509	0.541
Paragraph	0.944	0.925	0.964	0.969	0.805	0.728	0.825	0.841
Figure	0.855	0.854	0.917	0.965	0.674	0.609	0.754	0.797
Figure Caption	0.809	0.716	0.881	0.897	0.518	0.359	0.564	0.576
Table	0.864	0.824	0.919	0.941	0.668	0.602	0.748	0.761
Table Caption	0.763	0.590	0.891	0.903	0.424	0.317	0.519	0.524
Equation	0.857	0.825	0.875	0.920	0.652	0.521	0.635	0.719
Equation Number	0.832	0.594	0.890	0.916	0.565	0.122	0.486	0.657
Algorithm	0.368	0.231	0.463	0.665	0.327	0.173	0.406	0.527
Footnote	0.697	0.854	0.881	0.950	0.488	0.574	0.638	0.687
Page Number	0.519	0.346	0.630	0.670	0.206	0.098	0.216	0.261
Reference Heading	0.836	0.612	0.808	0.871	0.631	0.238	0.561	0.655
Reference Text	0.911	0.927	0.964	0.974	0.838	0.819	0.894	0.904
Combined (mAP)	0.774	0.596	0.832	0.886	0.573	0.356	0.590	0.655

Table 4: Object detection performance results.

lower quality (e.g., blurred) as compared to the training dataset used for other models. This results in lower overall performance of the model.

- **Size of the training dataset:** The **Scanned** model was trained on the smallest training dataset. Consequently, it has the lowest performance among all four variants. The large size of the training dataset used in **Digital + Scanned + AI-Aided** helps achieve the best overall performance.
- **Performance on minority classes:** We also find that training on a dataset with a better distribution in terms of object classes significantly improves performance. As can be seen from the results shown, the performance of certain categories, such as *Degree* and *Algorithm*, increased by ∼20%. This shows that model performance on certain low-performing categories can be improved by training on a larger number of samples from such categories.
- **Weak labels can be helpful signals for targeted annotation:** Another observation that can be made from the performance improvements achieved on low-frequency categories is that weak labels generated from an existing model can serve as a good indicator for more targeted annotation. Although using such labels cannot guarantee coverage, they can still address performance issues to a great extent.
- **Overall performance:** Finally, we can also observe that performance improvements are achieved in other categories that were not included in the filter set. This can be attributed to the fact that while the AI-Aided data consisted of pages filtered based on the occurrence of minority elements, these pages also contained other elements in addition to those from the filter set. This helped the model to be trained on more samples from other object categories as well, thus improving the performance across all object classes.

6 CONCLUSION AND FUTURE WORK

In this work, we address some of the major challenges related to layout analysis of long PDF documents such as theses and dissertations. To address the high costs of annotation due to the time required to annotate page images, we propose an AI-aided annotation framework. This framework utilizes the predictive power of models trained on smaller datasets to help with annotating page images for training object detection models. It can also help with addressing the class imbalance in object detection datasets, by filtering images for certain categories. These images can then be validated and corrected by human annotators before being used to train object detection models. Although we demonstrate the effectiveness of this technique on document-based datasets, we believe that the proposed approach can be used in other domains as well. Another contribution of this paper was ETD-ODv2, a dataset to help with object detection-based layout analysis of electronic theses and dissertations (ETDs). The data set consists mainly of manually annotated page images from scanned ETDs. It also consists of a subset sourced from both scanned and digital documents, designed to address the class imbalance problem in document-based object detection datasets. The dataset is compatible with existing datasets, allowing researchers to train object detection models on large training datasets.

In the future, we would like to extend this work to other types of academic documents, such as PDF slides. Our future work would also focus on making layout analysis compatible with document accessibility tools like on-screen readers, to benefit the wider community from our research. The code, datasets, and pre-trained models discussed in this paper are available at https://github.com/Opening-ETDs/ETD-OD.

ACKNOWLEDGMENTS

This project was made possible in part by the Institute of Museum and Library Services LG-37-19-0078-19. The authors thank the University Libraries and the Department of Computer Science at Virginia Tech for their generous support of this research. We also thank Andrew Leavitt, Annie Tran, Kecheng Zhu, Jiangyue Li, Zachary Gager, You Peng and Shelby Neal for their help in dataset curation.

REFERENCES

[1] Aman Ahuja, Alan Devera, and Edward Alan Fox. 2022. Parsing Electronic Theses and Dissertations Using Object Detection. In *Proceedings of the first Workshop on Information Extraction from Scientific Publications*. Association for Computational Linguistics, 121–130. https://aclanthology.org/2022.wiesp-1.14

[2] Frank Le Bourgeois, Zbigniew Bublinski, and Hubert Emptoz. 1992. A fast and efficient method for extracting text paragraphs and graphics from unconstrained documents. In *11th IAPR International Conference on Pattern Recognition, ICPR 1992. Conference B: Pattern Recognition Methodology and Systems, The Hague, Netherlands, August 30-September 3, 1992*. IEEE, 272–276. https://doi.org/10.1109/ICPR.1992.201771

[3] Kevin Dinh, Brian Dinh, Andrew Leavitt, and Annie Tran. 2022. Object Detection. http://hdl.handle.net/10919/114082 Virginia Tech CS4624 team term project.

[4] Ross B. Girshick. 2015. Fast R-CNN. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. IEEE Computer Society, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

[5] Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. LayoutLMv3: Pre-training for Document AI with Unified Text and Image Masking. In *MM '22: The 30th ACM International Conference on Multimedia, Lisboa, Portugal, October 10 - 14, 2022*. ACM, 4083–4091. https://doi.org/10.1145/3503161.3548112

[6] Zheng Huang, Kai Chen, Jianhua He, Xiang Bai, Dimosthenis Karatzas, Shijian Lu, and C. V. Jawahar. 2019. ICDAR2019 Competition on Scanned Receipt OCR and Information Extraction. In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*. IEEE, 1516–1520. https://doi.org/10.1109/ICDAR.2019.00244

[7] Guillaume Jaume, Hazim Kemal Ekenel, and Jean-Philippe Thiran. 2019. FUNSD: A Dataset for Form Understanding in Noisy Scanned Documents. In *2nd International Workshop on Open Services and Tools for Document Analysis, OSTICDAR 2019, Sydney, Australia, September 22-25, 2019*. IEEE, 1–6. https://doi.org/10.1109/ICDARW.2019.10029

[8] Sampanna Yashwant Kahu, William A. Ingram, Edward A. Fox, and Jian Wu. 2021. ScanBank: A Benchmark Dataset for Figure Extraction from Scanned Electronic Theses and Dissertations. In *ACM/IEEE Joint Conference on Digital Libraries, JCDL 2021, Champaign, IL, USA, September 27-30, 2021*. IEEE, 180–191. https://doi.org/10.1109/JCDL52503.2021.00030

[9] Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, Ming Zhou, and Zhoujun Li. 2020. TableBank: Table Benchmark for Image-based Table Detection and Recognition. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*. European Language Resources Association, 1918–1925. https://aclanthology.org/2020.lrec-1.236/

[10] Minghao Li, Yiheng Xu, Lei Cui, Shaohan Huang, Furu Wei, Zhoujun Li, and Ming Zhou. 2020. DocBank: A Benchmark Dataset for Document Layout Analysis. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. 949–960. https://doi.org/10.18653/v1/2020.coling-main.82

[11] Patrice Lopez and et al. 2008–2022. GROBID. https://github.com/kermitt2/grobid.

[12] Seunghyun Park, Seung Shin, Bado Lee, Junyeop Lee, Jaeheung Surh, Minjoon Seo, and Hwalsuk Lee. 2019. CORD: A Consolidated Receipt Dataset for Post-OCR Parsing. In *Workshop on Document Intelligence at NeurIPS 2019*. https://openreview.net/forum?id=SJl3z659UH

[13] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In *Advances in Neural Information Processing Systems 28: Annual Conference*

on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. 91–99. https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html

[14] Zejiang Shen, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. LayoutParser: A Unified Toolkit for Deep Learning Based Document Image Analysis. In *16th International Conference on Document Analysis and Recognition, ICDAR 2021, Lausanne, Switzerland, September 5-10, 2021, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 12821)*. Springer, 131–146. https://doi.org/10.1007/978-3-030-86549-8_9

[15] Tomasz Stanislawek, Filip Gralinski, Anna Wróblewska, Dawid Lipinski, Agnieszka Kaliska, Paulina Rosalska, Bartosz Topolski, and Przemyslaw Biecek. 2021. Kleister: Key Information Extraction Datasets Involving Long Documents with Complex Layouts. In *16th International Conference on Document Analysis and Recognition, ICDAR 2021, Lausanne, Switzerland, September 5-10, 2021, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 12821)*. Springer, 564–579. https://doi.org/10.1007/978-3-030-86549-8_36

[16] Sami Uddin, Bipasha Banerjee, Jian Wu, William A. Ingram, and Edward A. Fox. 2021. Building A Large Collection of Multi-domain Electronic Theses and Dissertations. In *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15-18, 2021*. IEEE, 6043–6045. https://doi.org/10.1109/BigData52589.2021.9672058

[17] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. 2022. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. *CoRR* abs/2207.02696 (2022). https://doi.org/10.48550/arXiv.2207.02696 arXiv:2207.02696

[18] Papers with Code. 2022. Real-Time Object Detection on COCO. https://paperswithcode.com/sota/real-time-object-detection-on-coco

[19] Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. LayoutLM: Pre-training of Text and Layout for Document Image Understanding. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*. ACM, 1192–1200. https://doi.org/10.1145/3394486.3403172

[20] Yang Xu, Yiheng Xu, Tengchao Lv, Lei Cui, Furu Wei, Guoxin Wang, Yijuan Lu, Dinei A. F. Florêncio, Cha Zhang, Wanxiang Che, Min Zhang, and Lidong Zhou. 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, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*. Association for Computational Linguistics, 2579–2591. https://doi.org/10.18653/v1/2021.acl-long.201

[21] Xu Zhong, Jianbin Tang, and Antonio Jimeno-Yepes. 2019. PubLayNet: Largest dataset ever for document layout analysis. In *2019 International Conference on Document Analysis and Recognition, ICDAR 2019, Sydney, Australia, September 20-25, 2019*. IEEE, 1015–1022. https://doi.org/10.1109/ICDAR.2019.00166

[22] Yichao Zhou, James B. Wendt, Navneet Potti, Jing Xie, and Sandeep Tata. 2022. Radically Lower Data-Labeling Costs for Visually Rich Document Extraction Models. *CoRR* abs/2210.16391 (2022). https://doi.org/10.48550/arXiv.2210.16391 arXiv:2210.16391

[23] Kecheng Zhu, Zachary Gager, Shelby Neal, Jiangyue Li, and You Peng. 2022. Object Detection. http://hdl.handle.net/10919/109979 Virginia Tech CS4624 team term project.