INTUITIVE

INnovative Network for Training in ToUch InteracTIVE Interfaces

Grant agreement: #861166 H2020-MSCA-ITN-2019

Start date: 2019-10-01 End date: 2023-09-30

Deliverable reporting document

Deliverable no: 5.1		WP: 5
Deliverable name: Taxonomy of image processing algorithms in the context of categorizing tactile graphics	Type: Report	Dissemination level: Public
Due Delivery date: 30 April 2022		Date delivered: 2 Mayl 2022

Description:

Systematic analysis of “translating and simplifying visual to tactile graphics” by a literature review.

Contents

Taxonomy of Image Processing Algorithms in the Context of Categorizing Tactile Graphics

Omar Moured (name.surname@kit.edu)

1 Abstract

In this project, we dedicate ourselves to the automatic preparation of graphical literature content for people with visual impairments or blindness. Visual representations of information are becoming more and more normal. Presenting information simply and compactly. However, people with visual impairment or even blindness have little or no access to graphic information. They are dependent on people describing graphic content to them or creating tactile variants. Content must become tactile and be explained. Lines, textures, structures, texts. In order to enable self-determined learning or study, for example, it is necessary to prepare the data in a complex way.

To automate this and improve access, many image processing algorithms nowadays make use of the advances in artificial intelligence in the field of computer vision. Therefore, in this work, the most important steps for extracting graphics from documents and the results achieved in publications so far were systematically examined in detail and the promising approaches were reproduced and tested with their own data.

Keywords tactile graphics/diagrams, computer vision, artificial intelligence.

2 Introduction

The sense of touch is an important source of data for individuals investigating scenes in nearby areas. It passes various tactile data. This sense is shown to be superior to the visual and auditory systems in perceiving accurate and complete characteristics of objects [1]. Nowadays, the amount of data available for browsing is nearly limitless. Thanks to the widespread use of the Internet and the passion of people to learn about their world. For visually impaired people, tactile graphics (TG) are an essential means of understanding the world and there is no doubt that they find it helpful, particularly in math and science [2].

According to the World Health Organization (WHO), there are about 246 million people with low vision impairment [3]. Many of these people live in low-income settings, and in some countries, visually impaired persons do not get financial support from the government. Hence, there has been towards the years an emphasis on more automated and affordable options for this group.

There are various ways to obtain TG. Various production technologies are available for this, e.g. using braille embossing machines [4], which can emboss graphics by punching rows of dots into a thicker paper. Embossing machines are very useful but have the double disadvantage that the output is static and the machines themselves are very expensive [4]. Alternatives are swell paper (expensive PVC paper) [5] or thermoforming (costly preparation of master artwork) [6]. More recently, 2D refreshable braille displays are used. Simply put, this is a matrix of dots that can dynamically change the displayed content via software or even allow interactive interaction with the content [7]. The one common characteristic between all these technologies is the creation process of the tactile content. While tactile graphics can be created manually by experts, the method is time-consuming. Therefore, in this report, I will focus on the taxonomies of automating the generation of TG and my detailed work on extracting the graphical and textual content from documents. The final goal is to enable users to work out and use graphic content on 2D Braille displays on their own.

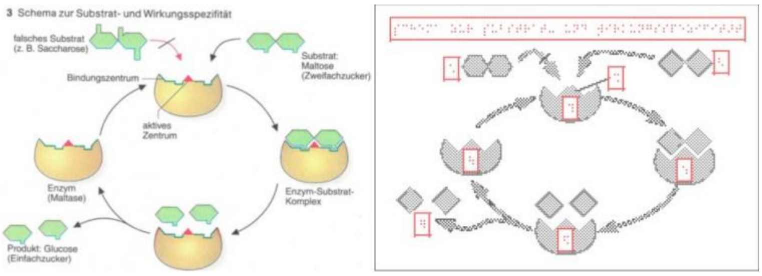
3 Schema zur Substrat- und Wirkungsspezifität
falsches Substrat (z. B. Saccharose)
Substrat: Maltose (Zweifachzucker)
Bindungszentrum
aktives Zentrum
Enzym (Maltase)
Enzym-Substrat-Komplex
Produkt: Glucose (Einfachzucker)

Figure 1. A sample of a scientific graphic from the biology process. On the right, the corresponding tactile version with title and labels in Braille is highlighted in red [8].

3 Graphics Types

TG are images designed to be touched rather than looked at and should deliver a piece of critical information to the user [9]. An example graphic is depicted in Figure 1. Since TG can convey information, there is a need to categorize the different types of visual graphics in order to make the automation process more targeted and thus more efficient. So far, there has been no attempt to identify them in the context of tactile graphics.

Therefore, we have tried to cluster the different types of graphics into groups, as shown in Table 1 below, to work on solutions in a targeted way. These types can thus also be considered as input for the image processing algorithms.

Table 1. Type of inputs to the image processing algorithms in the context of tactile graphics.

Super category	**Sub-categories**
Natural images	
Document graphics	General images
	Mathematical graphs
	Scientific graphics
	Maps
SVG images	Logos
	Symbols and signs
Visual artworks	
3D models	

The decision of super and sub-categories is based on the focus of the computer vision research community so far. In the table, natural images are those that depict the environment, they usually consist of complex objects, sometimes highly textured, buildings, trees, and so on [10]. SVG images are usually used to enhance the accessibility of web pages [11]. Visual artworks and 3D models are other interesting input types the community needs to investigate [12]–[14]. Yet, the major concern of visually impaired people nowadays is document graphics [15]. This concern can be understood because documents usually contain an intensive amount of information. In this project, we will concentrate on graphics within scientific articles and publications since it is quite available and is weakly approached by the research community due to their complex scientific content. We hope this work will inspire other researchers in the future and further explore this topic. Due to a large number of different categories and types of images, we will start by limiting the topic to mathematical graphs. More specifically, we will be processing line graphs.

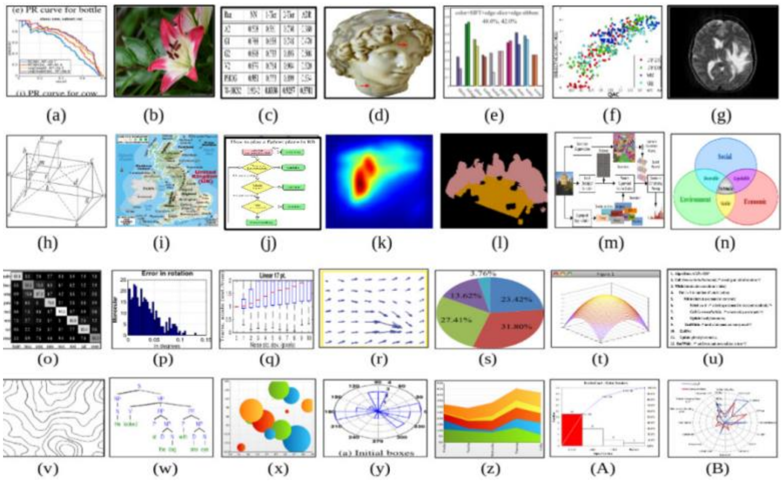

Figure 2. Visual illustration of the different graphics extracted from documents. (a) Line graph, (b) Natural image, (c) Table, (d) 3D object, (e) Bar plot, (f) Scatter plot, (g) Medical image, (h) Sketch, (i) Geographic map, (j) Flow chart, (k) Heat map, (l) Mask, (m) Block diagram, (n) Venn diagram, (o) Confusion matrix, (p) Histogram, (q) Box plot, (r) Vector plot, (s) Pie chart, (t) Surface plot, (u) Algorithm, (v) Contour plot, (w) Tree diagram, (x) Bubble chart, (y) Polar plot, (z) Area chart, (A) Pareto chart and (B) Radar chart. These images were collected from the DocFigure dataset [16].

3.1. Mathematical Graphs

Math is a fundamental subject not only as an academic discipline but also as a tool necessary in technology, science, and business. Statistical charts or graphs of mathematical equations are commonly used for the visual representation of data [17]. They are commonly used to deliver data that can be more easily understood rather than tables with numbers or plain text. Yet, till now the content of graphics in math and science books is not available in an accessible format. Because such graphics often lack appropriate alternative text (Alt-text), visually impaired users usually miss the information provided by such graphics since the content delivered by them is commonly not repeated in the article’s text [18]. For this reason, we have made the first attempt to enhance the accessibility of graphs by automatically extracting graphs from documents and then converting them into a tactile format, complemented by the automatic generation of alternative text. In this work, the focus will be to develop sustainable and practical solutions for accessible mathematical graphs, example categories are depicted in Figure 2. Since, for one thing, it is the common way used in STEM-related documents and On the other hand, artificial intelligence methods usually need a large amount of data for training, which we can easily collect from available open-source datasets such as Arxiv [19].

4 System Architecture

In the figure below we can see the flow diagram of the in-develop system, we are working on. The ultimate goal to be achieved is to automatically generate a TG, with a minimum number of interactions between the user and the software, along with the generation of a textual description which can be conveyed via screen reader but also as audio output. The scanned input document at the first step is fed to the Document Layout Analysing Module (DLAM) which was the focus of my first term within the project. In fact, DLAM is responsible for not only extracting elements but also localizing them pixel-wise for better post-processing later stages. As we will show later, our DLAM module is novel in terms of instance, elements of interest, and it is able to process. Those instances are shown in table 2. The instances will be extracted via region proposal in different formats as will be discussed later, in short, ensemble approach (bounding boxes and semantic segmentation). Pre-processing is necessary to overcome low-quality documents and prepare the input before the processing step. The fusion step of multiple outputs is required to obtain high confided detections. As we indicated earlier since only mathematical graphics are the object of interest, a pre-trained classifier to differentiate mathematical graphs is used. Another usage of the classifier is triggering the OCR engine inside the Context Module to minizine document processing time. When triggered text content is retrieved from proposed regions (paragraphs, captions, and graphs). The text classes such as title, paragraph, list and caption are directly obtained from the object detection model. Note that because the figure can be referenced later or on earlier pages within the document, I accumulate the converted text till the end page to make sure all figure references have been captured. Obtained text will be fed as an auxiliary element to the Natural Language Processing (NLP) model to generate a semi-structured description. Finally, at a later step of this project, I will investigate the graphs to tactile conversion module which should balance between graph tactile simplicity and generated alternative text.

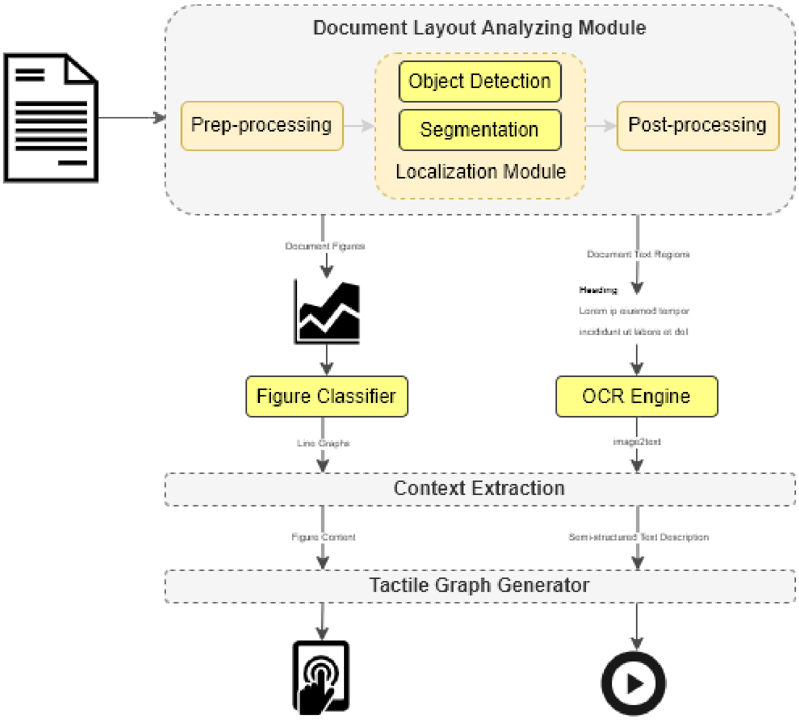

Figure 2. The in-development system architecture.

Table 2. Categories of the document instance our system can extract.

Document Instances	**Sub-Classes**
Figure	Mathematical graph
	Other
Text	Paragraph
	Caption
	Title
	List
	Other
Equation	-
Table	-

5 Document Layout Analysis

DLA is defined to be the most important step in the document analysis and recognition processes, it is the function of separating text, graphics, and images, and then extracting isolated text blocks (layout objects) such as titles, paragraphs, headers, footers, and captions [20]. Due to huge variation in document formats, textual and graphical structure, several challenges arise when extracting information:

- High variations in content. For example, various font styles, character sizes, multi-panel graphics like having multiple instances in one figure, and multi-column documents where characters have smaller sizes which makes them difficult to recognize. An example is shown in Figure 3 (a).
- Regions with overlapped boundaries. May limit image processing algorithm such as object detection where output bounding boxes are overlapping. As can be seen in Figure 3 (b), the paragraph shape is deformed due to another graphic. Since the object detection outputs rectangular bounding boxes, the regions as a result will overlap.
- Unaligned input documents. This issue is also like text with various orientations. Figure 3 (c) depicts an example scanned document with a wrong text alignment.
- Poor document quality. This is usually a concern if the input is obtained by scanning. In addition, the contained content within the document such as Images may be low resolution, causing noise and distortion.

Note that since we are working only with English documents, the language variations will not a major challenge for the DLA task.

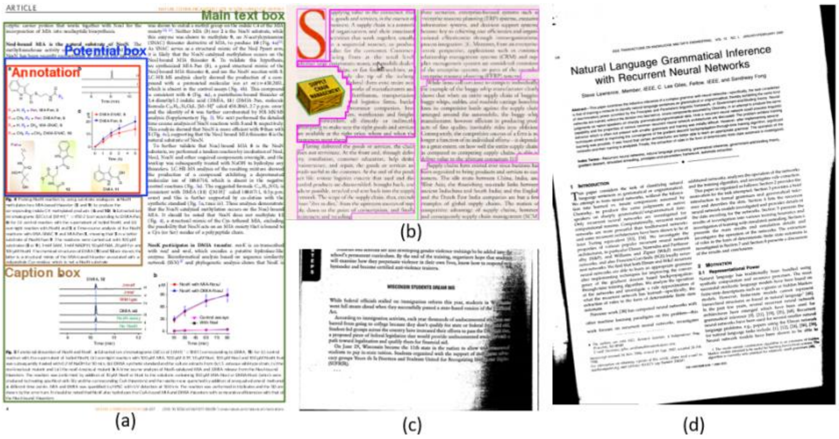
Main text box
Potential box
Annotation
Caption box
(a)
(b)
(c)
(d)
Natural Language Grammatical Inference with Recurrent Neural Networks

Figure 3. Example layout challenges for document analysis.

5.1. Background

According to my literature review, the research community approached the DLA task with four strategies: bottom-up, top-down, AI-based and hybrid. The bottom-up strategy is often data-dependent (in terms of algorithm parameters) [21]. It starts with layout analysis of small document elements such as pixels or connected components. Then, it merges homogeneous elements to create larger zones. It continues forming larger homogeneous regions until it reaches pre-defined stopping conditions. A top-down strategy starts from large document regions such as document-level. Then, it splits that large region into smaller zones such as text columns based on some homogeneity rules. The top-down analysis stops when there is no more splitting of zones, or some stopping conditions are reached [22]. AI-based methods can be viewed as either top-down or bottom-up methodologies. Because this approach uses either direct pixel-intensities or pixel features to identify zones and regions. Deep learning (DL) is quite common nowadays where trained models are utilized to extract deep layout features and map the model output as a detection or segmentation task [23]. The first two strategies are quite old compared to the recent advanced in computer vision (AI field). Usually, the old heuristics are utilized as an auxiliary path yielding what is called a hybrid strategy. In this work, the classical methods are only used up to now as pre-and post-processing steps to overcome naive challenges such as document alignment, refining classification and detection results.

5.2. Pre-processing

Document images may suffer from several degradations that negatively affect the performance of the DLA algorithms. Generally, there are two main sources of such degradations: native and auxiliary [24]. The former generated due to ageing, ink usage, writing style, etc and lead to layout issues such as text ink-bleeding, show-through, text fading, text-touching, text-spacing or baseline fluctuation will not be of this work interest. The latter is generated due to scanning-device malfunction such as failure in lighting conditions or misplacement of the document (document alignment). Such factors may lead to document image skew, blurring, black-edges, and the like. The negative effect of these issues has to be minimized before starting any layout analysis. Since the recent DL-based algorithms use only digital PDF documents, there was no emphasis on the pre-processing step. For the sake of complete accessibility of documents for the visually impaired people, we will extend the task to cover scanned documents as well

Skew Detection and Correction

As discussed above, Global document-image skew is formed because of auxiliary degradation. This procedure is tightly related to document segmentation. The failure of proper correction may lead to missing textual information or uninterpretable tactile graphics. To extract document regions, the input document images should be set at a standard form (i.e., 0° skew angle) [25]. There are several solutions discussed in the literature but for simplicity will only discuss the frequency domain approach which is currently in use.

Frequency Domain (Radon transform)

FD is among the earliest approaches that applied Fourier transform (FT) to detect/correct document image skew. The method proposed by. Lowther et al. start by accumulating local document blocks angles into an angular histogram. Then Radon Transform (RT) is used in multiple works to analyse the FT spectrum [26]. Due to minor peaks in the spectrum which may deviate correct skew angle, few heuristics were tested, and we found the best practice to roughly cluster the document component using a simple bottom-up or top-down approach. Next, RT is applied locally on the convex shapes that represent the document's blocks. Finally, a bootstrap aggregating (Bagging) is used to accumulate blocks' local skew-angles to compute the final document skew angle.

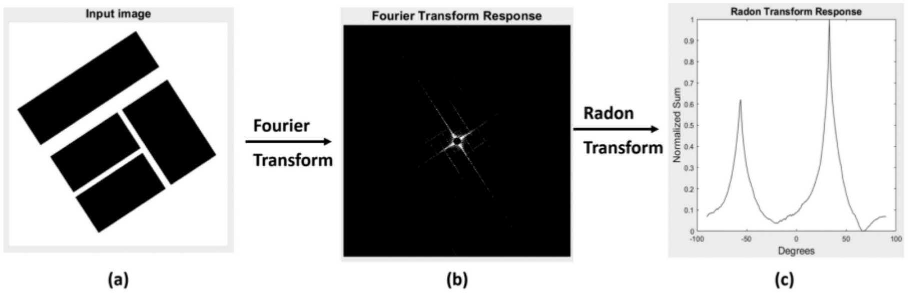

Figure 4. Frequency domain analysis: (a) Original document image, (b) Fourier Transform Magnitude, (c) Radon transform [26].

5.3. Content Localization Module

The first step toward recognizing a document element is localization. Recently, machine learning algorithms became dominant in solving pattern recognition and computer vision problems. This is due to the vast advances in computer technology that allowed fast processing and support larger memory capacities. Consequently, the learning-based methods get more attention to address complex layout analysis and derive both the logical and physical layout analyses. Unlike conventional machine-learning methods, usually deep learning methods generate features from image pixels for document layout analysis. However, most of them may require post-processing of the results. Another main concern is that deep learning models require a proper parameter initialization and usually need huge data to learn segmentation or classification tasks. Therefore, the training process takes a long time and large computation requirements.

Ensemble Mechanism

The ensemble is to combines several individual models to obtain better generalization performance [27]. In this work, I approached the aforementioned challenges through the Ensemble modelling approach where bounding boxes and pixel-level masks are obtained via object detection and semantic segmentation pre-trained models respectively. Then a fusion procedure is applied to combine the outcomes and enhance the performance. Since one of our main goals is to process scientific documents with possible overlapping structures, it is a necessity to utilize pixel-level masking. At the same time since we also intend to adapt not only digital documents but also scanned (captured) documents, as well as labelled documents for semination tasks, is poor, it is a necessity to utilize an object detector. The multimodal approach is a good way to combine the joint representations of different modalities. Yet, each modality can stand alone in some cases.

5.3.1. Object Detection

Object detection is the natural extension of object classification, which only aims at recognizing the objects in the image. The object detection model takes coloured image usually as input and output the bounding box coordinates of the objects of interest. The objects of interest are defined beforehand within a trained session [28]. The detector should be able to identify all instances of the object classes and draw a bounding box around it. It is generally seen as a supervised learning problem. Modern object detection models have access to large sets of labelled images for training and are evaluated on various canonical benchmarks. In the era of deep learning, detection algorithms can be split into two-stage [29] and the one-stage frameworks [30]. As shown in Figure 5, a network which has a separate module to generate region proposals is termed a two-stage detector, These models try to find an arbitrary number of object proposals in an image during the first stage and then classify and localize them in the second. As these systems have two separate steps, they generally take longer to generate proposals, have complicated architecture and lack global context. Single-stage detectors classify and localize semantic objects in a single shot using dense sampling. They use predefined boxes/keypoints of various scales and aspect ratios to localize objects. It edges two-stage detectors in real-time performance and is simpler. For this work, we will leverage the new detection model YOLOX.

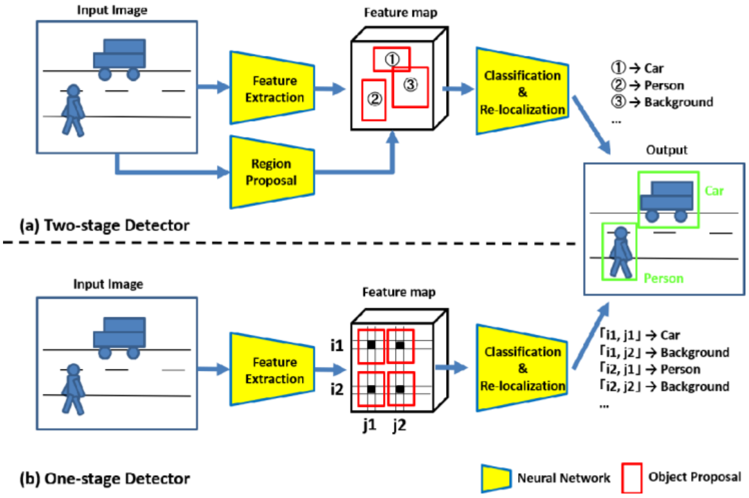

Figure 5. Illustration of the two-stage detector on top and one-stage detectors on the bottom.

YOLOX [30]

After multiple experimentation sessions on various state-of-the-art detection models, we decided to move forward with YOLOX. The reason for this choice is because of that YOLOX is anchor free method. Anchor boxes are a set of predefined bounding boxes of a certain height and width. These boxes are defined to capture the scale and aspect ratio of specific object classes of concern and are typically chosen based on object sizes in the targeted training datasets [31]. Anchor free mechanism significantly reduces the number of design parameters. The anchor mechanism has many known problems. First, to achieve optimal detection performance, one needs to conduct a clustering analysis to determine a set of optimal anchors before training. Those clustered anchors are domain-specific and less generalized. Second, the anchor mechanism increases the complexity of detection heads, as well as the number of predictions for each image. On some edge AI systems, moving such a large number of predictions between devices (e.g., from NPU to CPU) may become a potential bottleneck in terms of the overall latency. Furthermore, we tuned the training parameters of YOLOX to match the default dimensions of documents (A4). This tunning made the detection of very big and very small instances possible.

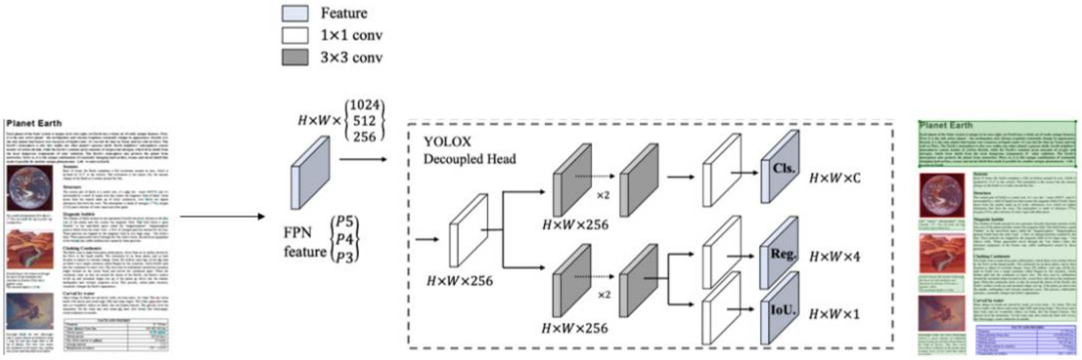

Figure 6. The model architecture of YOLOX with an example input-output image.

5.3.2. Semantic Segmentation

The task of document object detection is more focused on localizing layout elements of a document image such as tables, paragraphs, titles, figures etc [32]. with bounding boxes and then predicting the semantic category for that particular object. For complex document layouts, extracting the layout structure with only bounding box information becomes quite difficult in the case of overlapped regions. The segmentation task provides a more precise way of extracting structured layouts, as it involves the accurate prediction of masks (instances) at the pixel level. Modern successful approaches to solve the task are mostly based on CNN Although CNN helps to capture local features (low-level semantics) quite effectively from images, it performs sub-optimally in getting global semantic reasoning of features at the high-level [33]. Also, these methods follow a top-down approach which is highly reliant on the object detector to detect regions with bounding boxes, and then predict per-pixel segmentation masks. This turns out to be a more computationally expensive strategy in terms of inference speed and also inferior segmentation performances in more complex structured layouts.

SegFormer [34]

Not to increase the complexity of the overall system, Segformer, a recent simple, efficient yet powerful semantic segmentation framework is adapted. SegFormer unifies Transformers with a lightweight multilayer perceptron (MLP) decoder. Besides leveraging transformer layers, multiscale coarse-to-fine features are computed from the input to support an inference of multi-resolution inputs when deploying the model. Another advantage is that SegFormer eliminated the need for positional encoding, which degrades the performance in other sequence models. Our contribution to the SegFormer method is adding multi labelling assignments for pixels. Simplifying scene semantics with single-class labels might be sufficient for the most popular use-cases in scene segmentation, i.e., autonomous driving, but multi-label semantic segmentation is a must in the researched task of digitalizing documents. Imagine a figure of a map with text indicating countries. In this case, the conventional segmentation methods would assign a single label (text or figure) to that pixel while obtaining both classes would be more meaningful for this task. This contribution is achieved by slightly modifying the output mapping module of the SegFormer. Further studies and experiments were conducted to improve the outcomes' performance.

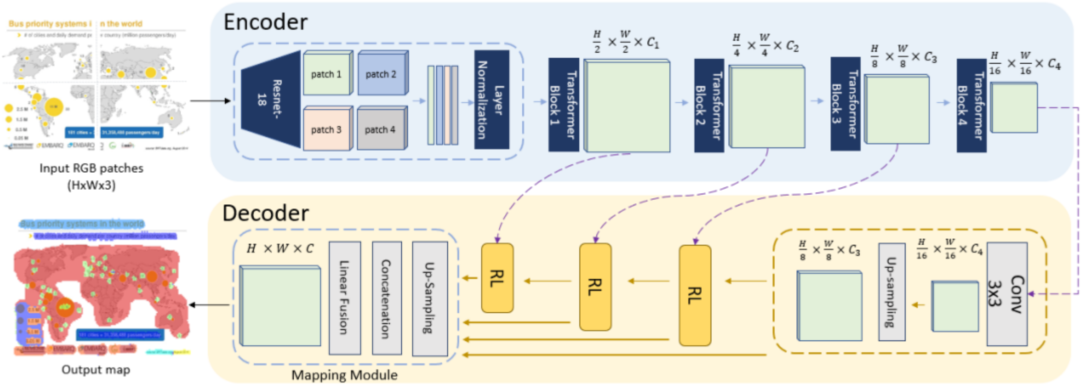
Encoder
Bus priority systems in the world
Input RGB patches
(HxWx3)
Resnet-18
patch 1
patch 2
patch 3
patch 4
Layer Normalization
Transformer Block 1
$\frac{H}{2} \times \frac{W}{2} \times C_1$
Transformer Block 2
$\frac{H}{4} \times \frac{W}{4} \times C_2$
Transformer Block 3
$\frac{H}{8} \times \frac{W}{8} \times C_3$
Transformer Block 4
$\frac{H}{16} \times \frac{W}{16} \times C_4$
Decoder
Output map
$H \times W \times C$
Linear Fusion
Concatenation
Up-Sampling
Mapping Module
RL
RL
RL
$\frac{H}{8} \times \frac{W}{8} \times C_3$
Up-sampling
$\frac{H}{16} \times \frac{W}{16} \times C_4$
Conv 3x3

Figure 7. The architecture of the semantic segmentation model, SegFormer [34], with an example input-output.

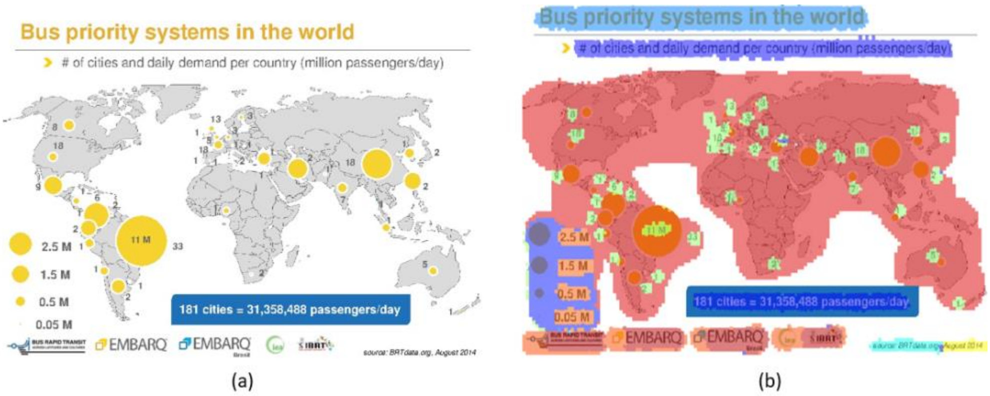
Bus priority systems in the world
# of cities and daily demand per country (million passengers/day)
11 M
2.5 M
1.5 M
0.5 M
0.05 M
181 cities = 31,358,488 passengers/day
EMBARQ
source: BRTdata.org, August 2014
(a)
(b)

Figure 7. Example of a map graphic on the left with multiple overlapping classes such as text and map pixels. On the right is a sample outcome result. Note that the text inside the map consists of two labels which are indicated as a single colour (green).

5.4. Fusion and Post-processing

Once previous methods complete the extraction phase, some post-processing steps are required. These steps help in overcoming performance degradation in particular cases, insure and emphasise method scalability to cover various document layouts and also compensate for object detection or segmentation limitations. We first adapted Syncretic-NMS to fit detections to objects. Furthermore, bounding boxes are also refined if the computer semantic masks are highly confident. Keep in mind the superior method here is object detection. The segmentation model is a way to overcome missed detections and utilize multi-labels when necessary.

Syncretic-NMS [35]

To assure proper fusion of bounding boxes with binary masks without degrading the performance, we adapted the concept of Non-Maximum-Suppression [36]. NMS is a procedure to select one entity (e.g., bounding boxes) out of many overlapping entities. But the goal is to keep only highly confident true positives. The conventional NMS is simple and intuitive: For a set of overlapping bounding boxes, the bounding box with the maximum score is selected, and the neighbouring bounding boxes are deleted according to specified rules. In this case, if they exceed the manually set IOU (intersection over union) threshold. Due to the conciseness of these conditions, traditional NMS has extremely high efficiency, but some high confidence bounding boxes may also be filtered out by mistake, thereby resulting in the obtained bounding boxes not including complete objects. Syncretic-NMS uses the same procedure as NMS but instead of coarse comparison and elimination of the bounding boxes, it smoothly merges the high confident neighbouring boxes. It judges the neighbouring bounding boxes of each bounding box and combines the neighbouring boxes that are strongly correlated with the corresponding bounding boxes. The coordinates of the merged box are the four coordinate extremes of the bounding box and the highly relevant neighbouring box. The authors also proposed a method to tune the threshold to filter out neighbouring objects.

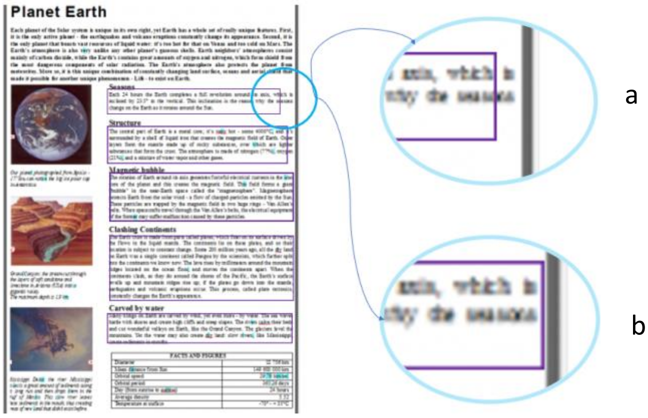

Figure 8. A false detection bounding boxes that do not fit the object. (a) the conventional result with NMS. (b) The enhanced coordinates after utilizing Syncretic-NMS.

5.5. Figures Classification

Documents contain various types of figures (e.g. Bar charts, Pie charts, Line plots, etc.) to present heterogeneous information in a compact and visual form. This visual representation of complex information helps the reader to easily understand the content of the documents [16]. A better understanding of documents also requires an understanding of the figures present in the documents. Since the automatic understanding of these figures is the optimal goal of this work, the classification of document figures into various categories is a necessity as an initial task. Classification of document figures is also a complex task due to inter-class visual similarity and intra-class visual dissimilarity among figures. Due to the aforementioned challenges, we adapted the largest document figures datasets. Namely, DocFigure. In the next section, all utilized datasets will be elaborated on in detail.

ResNet-101 [37]

ResNet is a deep learning convolutional neural network that achieved state-of-the-art image classification tasks. The motivation behind the ResNet is modifying the conventional Artificial Neural Networks with more profound layers with high exactness. The idea is to integrate "alternative routes" for the feedforward deep features and also weight updating. These identical shortcut connection helps achieve faster training and maintain a high performance [37]. In this work, we trained a ResNet with 101 layers (ResNet-101) to predict among 28 different categories. After experimenting with document figures, only predicted line graphs are then fed to the context and conversion module.

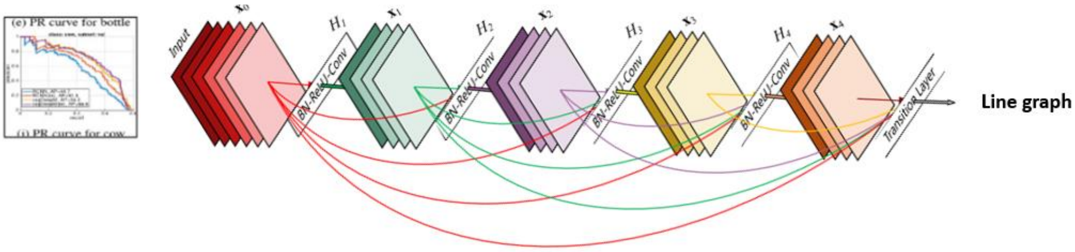

Figure 9. The architecture of the ResNet model is used for the figure classification task [38].

6 DATASETS

The main concern of any deep learning related work is the dataset. That is why a good amount of time is spent at the beginning to collect meaningful and comprehensive data for the task of "line graphs understanding and conversion to tactile display". We categorized the datasets in this work into two categories, private and public.

6.1. Private Data

In the ACCESS@KIT lab, we have implemented a portal where teachers mainly and interested local users can access and convert their scientific documents to an accessible format. The platform had auto-labelling features and also manual modification to solve wrongly processed elements. Within the past two years, a large amount of teaching material data was collected. Around 2 GBs of data and 10K documents were counted and used to train the YOLOX detector. Unfortunately, segmentation masks were not available due to the cost and time required for labelling. Instead, publicly available data were chosen to train the segment.

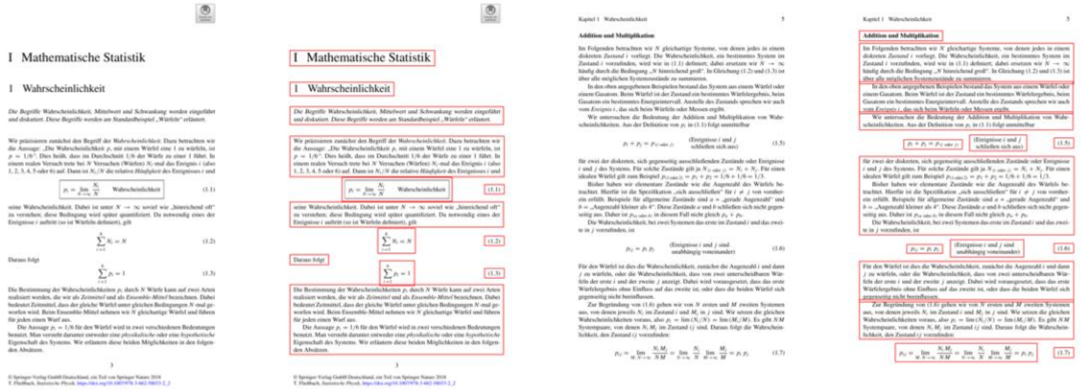
I Mathematische Statistik
1 Wahrscheinlichkeit
I Mathematische Statistik
1 Wahrscheinlichkeit
Addition und Multiplikation
Addition und Multiplikation

Figure 10. Example document with ground-truth labelling from ACCESS@KIT dataset.

6.2. Public Data

6.2.1. PubLayNet [39]

it is an image-based document layout dataset. It is one of the currently largest ever available document layout datasets. It consists of annotations of 5 categories on 358,353 pages, which are split into 335,703, 11,245 and 11,405 pages for training, validation, and testing, respectively. The reason we chose this dataset is that it has 96,656 annotated pages which contain at least one figure. Approximately it offers a 125k bounded figure. PubLayNet also provided semantic maps which made the training of SegFormer possible. Table 3 below shows the dataset's categorises. Although they are slightly different from Table 1, we used this data to enlarge the overall overlapped categories. Note that because PubLayNet only picked documents that are related to the medical field we don't have an equation category.

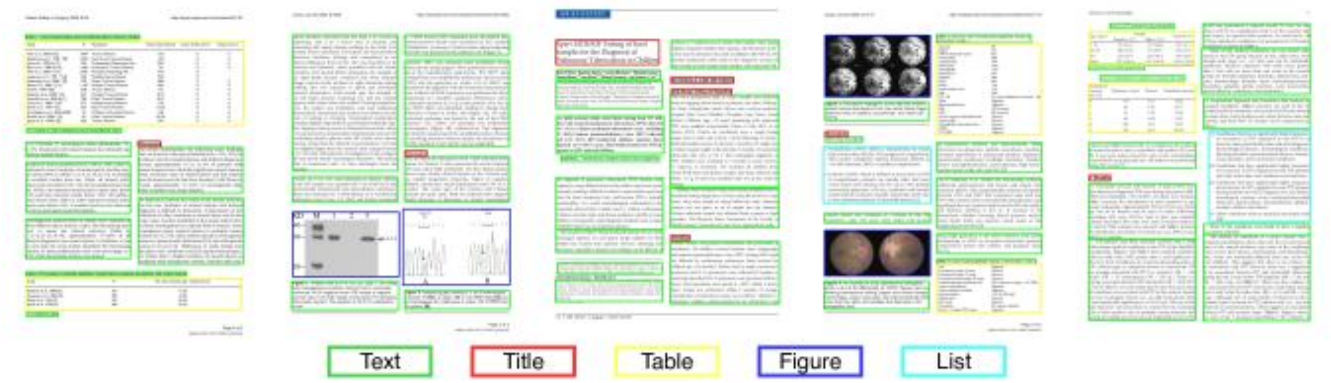

Figure 11. Sample images from PubLayNet dataset with coloured ground-truth labelling.

6.2.2. DocBank [40]

The currently largest document-level benchmark with fine-grained token-level annotations for layout analysis. It consists of 500k annotated LaTeX pages provided in different formats, images for detection models and text for NLP and OCR. semantics were also labelled in addition to bounding boxes. It split the data into 14 categories shown in table 3 below. Unlike PubLayNet, comprehensive categorise are provided by the DocBank dataset and it matches our selection except for some categories which are merged.

Table 3. DocBank dataset categories.

Semantic categories of the DocBank dataset						
Abstract	Author	Caption	Date	Equation	Figure	Footer
Table	Title	List	Paragraph	Reference	Section	

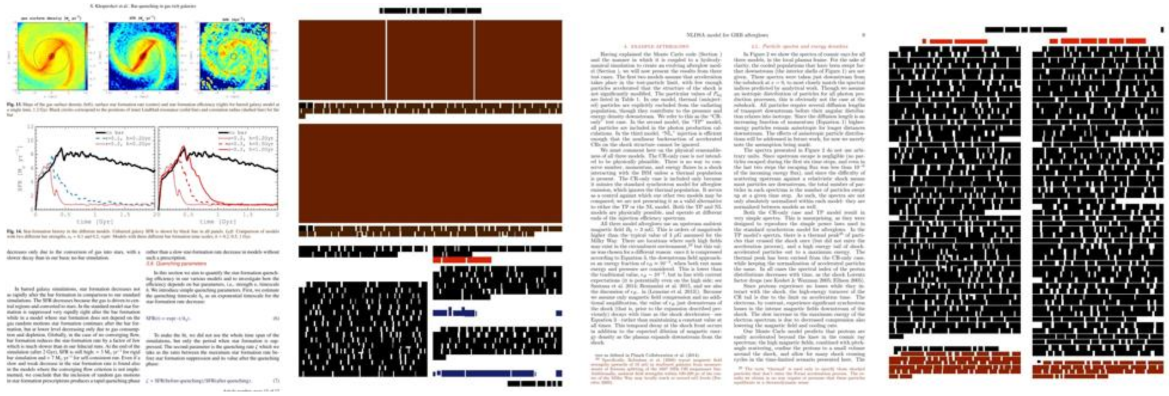

Figure 12. Sample documents with ground truth as masks from the DocBank dataset [40].

7 Discussion & Future Work

Within only the first 6 months of my start date and up to this point, we managed to extract line graphs and text from documents. Multiple deep learning-based and classical heuristics were used to achieve this task efficiently. The overall contributions so far can be summarised in the following points:

- First attempt to identify graphics in the context of tactile graphics for the computer vision community.
- expansion of the literature direction to include not only digital PDF documents but also scanned ones (image-based input). This includes the development of the pre-processing module.
- Utilization of Ensemble Mechanism to enrich the targeted domain of the system and to overcome challenging cases.
- YOLOX Tunning to handle the very small instances within documents. Especially, the multi-column-based ones.
- Major changes in the state-of-the-art segmentation model, SegFormer, to handle the multi-label assignment for pixels.

It is now our focus to extract the contextual features and generate a semi-structure description out of line graphs. Furthermore, since the datasets used so far targeted specific domains, we are planning to expand the variations of our training and testing batches by using ArXiv open-source bulk datasets. These datasets will be of great interest to the computer vision community and will be the building block for other state-of-the-art achievements. ArXiv documents are provided as LaTeX files, hence, an automatic labelling procedure can be used to obtain with ease the ground-truth data. Towards the end of the year, the tactile conversion module will be investigated. Yet, if time permits at the end f the project the following points will be revisited:

- Instead of a split detection-segmentation mechanism. We will try to experiment with a single model with a dual-head achieving the same task to reduce the complexity.
- Enhance the pre-processing step via integrating a super-resolution module to overcome blurred documents and sharpen the scanned ones.
- Instead of having multiple detection models trained on different datasets, due to different categorizations, we would like to investigate the merging of these datasets in a time-efficient way. One way is to use a pre-trained classifier on text classes from the DocBank dataset and then use it as an auto-labelling for PubLayNet. Another idea is to use so-called cross-dataset knowledge transfer. With this approach, we will reduce the overall computational complexity of the system and hence the inference time is reduced.
- Working on automatic labelling of ArXiv bulk documents to enrich the research community with valuable variations of documents with compressive categories.
- Instance-level semantics can be generated to distinguish the inter-class instances such as separated paragraphs or figures within a figure.

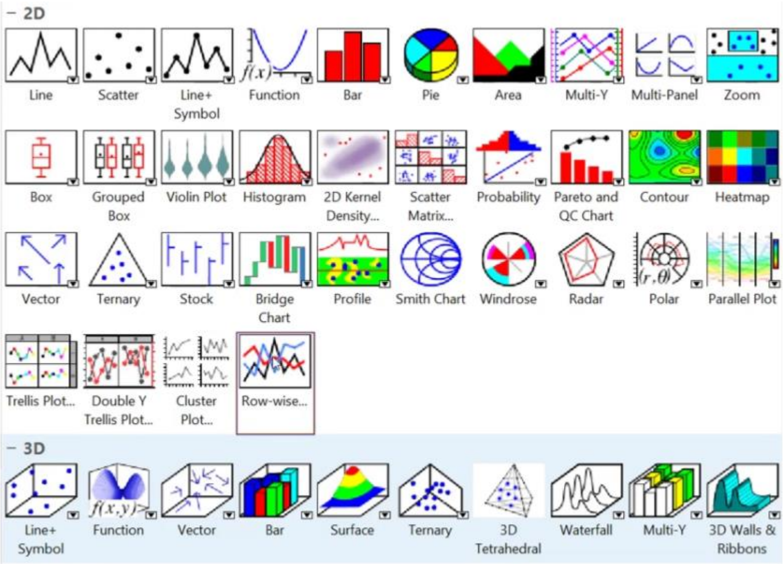

Figure 13. Different types of mathematical graphics [41]. Only line graphics will be considered in this work.

References

[1] S. Luo, J. Bimbo, R. Dahiya, and H. Liu, “Robotic tactile perception of object properties: A review,” *Mechatronics*, vol. 48, pp. 54–67, Dec. 2017, DOI: 10.1016/J.MECHATRONICS.2017.11.002.

[2] K. T. Zebehazy and A. P. Wilton, “Straight from the Source: Perceptions of Students with Visual Impairments about Graphic Use”, Accessed: Apr. 26, 2022. [Online]. Available: http://jvib.org/CEUs.

[3] “Blindness and vision impairment.” https://www.who.int/en/news-room/fact-sheets/detail/blindness-and-visual-impairment (accessed Apr. 26, 2022).

[4] M. Mukhiddinov and S. Y. Kim, “A Systematic Literature Review on the Automatic Creation of Tactile Graphics for the Blind and Visually Impaired,” *Processes 2021, Vol. 9, Page 1726*, vol. 9, no. 10, p. 1726, Sep. 2021, doi: 10.3390/PR9101726.

[5] “School Health Swell Touch Paper.” https://www.schoolhealth.com/swell-touch-paper (accessed Apr. 29, 2022).

[6] “ITD Journal: Tactile Graphics: An Overview And Resource Guide.” http://itd.athenpro.org/volume3/number4/article2.html (accessed Apr. 29, 2022).

[7] W. Yang, J. Huang, R. Wang, W. Zhang, H. Liu, and J. Xiao, “A Survey on Tactile Displays for Visually Impaired People,” *IEEE Transactions on Haptics*, vol. 14, no. 4, pp. 712–721, 2021, doi: 10.1109/TOH.2021.3085915.

[8] G. Melfi, K. Müller, T. Schwarz, G. Jaworek, and R. Stiefelhagen, “Understanding what you feel: A Mobile Audio-Tactile System for Graphics Used at Schools with Students with Visual Impairment,” *Conference on Human Factors in Computing Systems - Proceedings*, Apr. 2020, doi: 10.1145/3313831.3376508.

[9] L. P. Rosenblum and T. S. Herzberg, “Braille and Tactile Graphics: Youths with Visual Impairments Share Their Experiences”, Accessed: Apr. 26, 2022. [Online]. Available: http://jvib.org/CEUs.

[10] N. Ezaki, M. Bulacu, and L. Schomaker, “Text detection from natural scene images: Towards a system for visually impaired persons,” *Proceedings - International Conference on Pattern Recognition*, vol. 2, pp. 683–686, 2004, doi: 10.1109/ICPR.2004.1334351.

[11] M. Rotard, K. Otte, and T. Ertl, “Exploring Scalable Vector Graphics for Visually Impaired Users,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3118, pp. 725–730, 2004, doi: 10.1007/978-3-540-27817-7_108.

[12] L. Lydy Reidmiller, “ART FOR THE VISUALLY IMPAIRED AND BLIND: A CASE STUDY OF ONE ARTIST’S SOLUTION ,” 2003. https://etd.ohiolink.edu/apexprod/rws_etd/send_file/send?accession=osu1054144608&disposition=inline (accessed Apr. 29, 2022).

[13] B. Biggs, J. M. Coughlan, and P. Coppin, “Design and evaluation of an interactive 3D map,” *Rehabilitation Engineering and Assistive Technology Society of North America*, vol. 2021, 2021, Accessed: Apr. 29, 2022. [Online]. Available: /pmc/articles/PMC8341294/

[14] E. Lin, N. Labib, and J. Meyer, “3D Printed Accessible Map,” *Symposium of Student Scholars*, Apr. 2022, Accessed: Apr. 29, 2022. [Online]. Available: https://digitalcommons.kennesaw.edu/undergradsymposiumksu/spring2022/presentations/360

[15] S. Murphy, “Accessibility of Graphics in Technical Documentation for the Cognitive and Visually Impaired,” *Proceedings of the 23rd annual international conference on Design of communication documenting & designing for pervasive information - SIGDOC ’05*, 2005, doi: 10.1145/1085313.

[16] K. v. Jobin, A. Mondal, and C. v. Jawahar, “DocFigure: A dataset for scientific document figure classification,” *2019 International Conference on Document Analysis and Recognition Workshops, ICDARW 2019*, vol. 1, pp. 74–79, Sep. 2019, doi: 10.1109/ICDARW.2019.00018.

[17] C. Jayant and C. Jayant, “A Survey of Math Accessibility For Blind Persons and An Investigation on Text/Math Separation,” 2006, Accessed: Apr. 29, 2022. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.187.9287

[18] P. Moraes, G. Sina, K. Mccoy, and S. Carberry, “Evaluating the Accessibility of Line Graphs through Textual Summaries for Visually Impaired Users”, doi: 10.1145/2661334.2661368.

[19] “arXiv Bulk Data Access | arXiv e-print repository.” https://arxiv.org/help/bulk_data (accessed Apr. 28, 2022).

[20] “Document Layout Analysis Based on Emergent Computation | Proceedings of the 4th International Conference on Document Analysis and Recognition.” https://dl.acm.org/doi/10.5555/646270.684836 (accessed Apr. 27, 2022).

[21] L. Cui, Y. Xu, T. Lv, and F. Wei, “Document AI: Benchmarks, Models and Applications,” Nov. 2021, doi: 10.48550/arxiv.2111.08609.

[22] X. Wu, Z. Hu, X. Du, J. Yang, and L. He, "DOCUMENT LAYOUT ANALYSIS VIA DYNAMIC RESIDUAL FEATURE FUSION," *Proceedings - IEEE International Conference on Multimedia and Expo*, 2021, doi: 10.1109/ICME51207.2021.9428465.

[23] H. T. Tran, N. Q. Nguyen, T. A. Tran, X. T. Mai, and Q. T. Nguyen, "A Deep Learning-Based System for Document Layout Analysis," *2022 The 6th International Conference on Machine Learning and Soft Computing*, pp. 20–25, Jan. 2022, doi: 10.1145/3523150.3523154.

[24] G. M. Binmakhashen and S. A. Mahmoud, "Document Layout Analysis," *ACM Computing Surveys (CSUR)*, vol. 52, no. 6, Oct. 2019, doi: 10.1145/3355610.

[25] R. Ahmad, S. Naz, and I. Razzak, "Efficient skew detection and correction in scanned document images through clustering of probabilistic hough transforms," *Pattern Recognition Letters*, vol. 152, pp. 93–99, Dec. 2021, doi: 10.1016/J.PATREC.2021.09.014.

[26] G. Meng, C. Pan, N. Zheng, and C. Sun, "Skew estimation of document images using bagging," *IEEE Transactions on Image Processing*, vol. 19, no. 7, pp. 1837–1846, Jul. 2010, doi: 10.1109/TIP.2010.2045677.

[27] M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan, "Ensemble deep learning: A review," Apr. 2021, doi: 10.48550/arxiv.2104.02395.

[28] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, vol. 126, p. 103514, Jun. 2022, doi: 10.1016/J.DSP.2022.103514.

[29] "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." https://proceedings.neurips.cc/paper/2015/hash/14bfa6bb14875e45bba028a21ed38046-Abstract.html (accessed Apr. 29, 2022).

[30] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "YOLOX: Exceeding YOLO Series in 2021," vol. 5, p. 12, Jul. 2021, doi: 10.48550/arxiv.2107.08430.

[31] "Anchor Boxes for Object Detection - MATLAB & Simulink." https://www.mathworks.com/help/vision/ug/anchor-boxes-for-object-detection.html (accessed Apr. 29, 2022).

[32] Y. Mo, Y. Wu, X. Yang, F. Liu, and Y. Liao, "Review the state-of-the-art technologies of semantic segmentation based on deep learning," *Neurocomputing*, Jan. 2022, doi: 10.1016/J.NEUCOM.2022.01.005.

[33] S. Biswas, A. Banerjee, J. Lladós, and U. Pal, "DocSegTr: An Instance-Level End-to-End Document Image Segmentation Transformer," Jan. 2022, doi: 10.48550/arxiv.2201.11438.

[34] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers," May 2021, doi: 10.48550/arxiv.2105.15203.

[35] J. Chu, Y. Zhang, S. Li, L. Leng, and J. Miao, "Syncretic-NMS: A Merging Non-Maximum Suppression Algorithm for Instance Segmentation," *IEEE Access*, vol. 8, pp. 114705–114714, 2020, doi: 10.1109/ACCESS.2020.3003917.

[36] H. Zhang and X. Hong, “Recent progresses on object detection: a brief review,” *Multimedia Tools and Applications*, vol. 78, no. 19, pp. 27809–27847, Oct. 2019, doi: 10.1007/S11042-019-07898-2/TABLES/3.

[37] S. Targ, D. Almeida, and K. L. Enlitic, “Resnet in Resnet: Generalizing Residual Architectures,” Mar. 2016, doi: 10.48550/arxiv.1603.08029.

[38] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-December, pp. 770–778, Dec. 2016, doi: 10.1109/CVPR.2016.90.

[39] X. Zhong, J. Tang, and A. J. Yepes, “PubLayNet: largest dataset ever for document layout analysis,” *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*, pp. 1015–1022, Aug. 2019, doi: 10.48550/arxiv.1908.07836.

[40] M. Li *et al.*, “DocBank: A Benchmark Dataset for Document Layout Analysis,” pp. 949–960, Jun. 2020, doi: 10.48550/arxiv.2006.01038.

[41] “Origin: Data Analysis and Graphing Software.” https://www.originlab.com/origin (accessed Apr. 28, 2022).