University of Sussex

Document Segmentation and Classification using Yolo-V4

Supervised by

Dr Adam Barett

In partial fulfillment
of the requirement for the degree of

Department of Informatics:

BSc (Hons) Computer Science 2020/21

By
Harteley Sebastian
Candidate Number: 215905

2021 May 16

Statement of Originality

This report is submitted as part requirement for the degree of BSc (Hons) Computer Science at the University of Sussex. It is the product of my own labour except where indicated in the text. The report may be freely copied and distributed provided the source is acknowledged. I hereby give / withhold permission for a copy of this report to be loaned out to students in future years.

Signed by,
Harteley

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere gratitude to my senior supervisor, Dr Adam Barett for his patience, motivation and immense knowledge. Your guidance has helped me in completing my dissertation.

Secondly, I am also grateful to Dr Dmitrijs Dmitrenko and Dr Imran Khan, the Computer Science Project module leader and coordinator. I am thankful to them for sharing expertise, valuable guidance, and encouragement extended to me.

Third, I would like to thank all of my family and friends who supported me throughout the completion of my project. Their kind cooperation has allowed me to finish this research.

Also, I would like to thank University of Sussex and the lecturers who have taught me throughout my degree program here. They have instilled in me the wisdom and integrity to face any challenges as well as constantly striving for excellence.

Last but not the least, all of my colleagues at University of Sussex for providing me the warm and friendly atmosphere.

ABSTRACT

This work is focused on segmenting and classifying the physical layout of the document using object detection methods and techniques with contextual features. One major challenge in literature mining is the difficulty of detecting the region and extracting the detected region into its proper format from PDF files. For this, I proposed object detection techniques for document layout detection and classification and incorporate contextual information to also improve the region detection performance using the You Only Look Once (Yolo) object detection deep learning model.

Artificial intelligence, as we all know, is changing every aspect of life. Hence, using AI model to segmenting document section and detection of the segmented sections to their classes, such as title, author, figure, etc. The system will classify the documents on the base of the texts and figures and shows the detected sections with bounding boxes and classify them.

Deep learning is a sub-field of Artificial Intelligence that has shown impressive results in recent years. The aim of this study is to develop a simple document segmentation and classification system that is developed using Convolutional Neural Networks (CNN), one of deep-learning techniques, such as You Only Look Once Version 3 (Yolo-V3) and You Only Look Once Version 4 (Yolo-V4) for object detection and classification. Comparing the results of those models to each other.

The most common Deep-learning frameworks and libraries, such as Tensorflow, Keras, and OpenCV, are used to create deep learning models like CNN. They're commonly used in real-time apps and applications. In addition, the proposed model includes end-to-end applications that searches each document before detecting and segmenting regions.

In this work, a document physical layout segmentation detection and classification base system is designed to help the document analysis for literature mining more effectively. The proposed model outperforms the baseline method such as Region Based Convolutional Neural Networks (R-CNN) in terms of performance.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Documents are considered as important sources of knowledge and information for various cognitive processes such as Optical Character Recognition (OCR), documents segment, graphic understanding, and many more [4].

This work focuses on segmenting and classifying the physical layout of the document using object detection methods and techniques with contextual features. Document segmentation and classification is a process of subdividing document regions and categorizing the regions into groups. Since millions of records, such as government archives, technical reports, bank checks, magazines, books, letters, newspapers, and so on, must be processed every day [53] and the number and variety of scientific literature grows each day. The abundance of archived scientific literature provides a detailed foundation for future development. Literature mining aims to find knowledge embedded in scientific literature that is not available using traditional statistical methods [49]. That is why if document analysis can be done automatically, it will save a lot of effort and time. The automation of document analysis entails extracting regions, identifying the type of the segmented area, and then dealing with each region separately [53]. One major challenge in automation of document analysis is the difficulty of detecting the region and identifying the region from PDF files. For this, I proposed object detection techniques for document layout detection and classification and incorporate contextual information to also improve the region detection performance using the You Only Look Once (Yolo) object detection deep learning model.

This discussion provides a concrete foundation for research activities in this study by including a full understanding and a brief detail of the project. In addition, a brief discussion of the problem statement is given in order to gain a thorough understanding of the issue.

1.1 Background of the Project

A digital document is a valuable source of data for a variety of cognitive processes such as Optical Character Recognition(OCR), knowledge database creation, documents retrieval, graphic understanding, text extraction, and others. One of the major challenges is that to extract information and segment out the region digitally .which consists of categorization and identifying the documents images ROI(Region of Interest).

In past literature work, various methods has been adopted for detection and classification of documents layout , and In [65], author proposed some techniques

to classify the documents into various groups (i) pixel-based classification methods [40], [41].(ii) region-based on black-based classification documents [31], [52], (iii) connect component classification methods [32],[30] and [62]. The method based on the block or region-based classification segments out the images of documents into document zones and then using different techniques they classify those documents into their representative of the semantic classes. In pixel-based classification methods, the other proposed the technique by taking each of the individual pixels into calculation and using a classifier to create a labeled picture with hypotheses of the regions. And also a connected components method uses refined and local combined and finally classified.

On the other hand when discussing image classification, what comes to mind is CNN (convolution Neural network). It is a network that has an end-to-end automatic feature extraction and classification, which is why it has been widely used in many for various tasks, including documents analysis [37], [5]. However, the cost of using them in storage of documents and applications for retrieval is normally limited due to their inherent high computing load. [47], [48] proposed a model that reduces the computational load of documents analysis for identifying image blocks by using projections, but on the other hand, the robustness of CNNs using a 1-D convolutional network (Conv1D) which is a layer convoluting the layer input with the convolution kernel over a single spatial (or temporal) dimension yields a tensor of outputs, is affected not in a good way. This method has created various opportunities to form a novel CNN-based network for document segmentation and analysis that reduces computational costs and data use without compromising accuracy.

In this work, I proposed effective techniques for document segmentation and classification of the regions using object detection in real-time techniques, which is called Yolo (You Only Look Once). Using Yolo models for document segmentation and classification of sections can be useful for identifying the regions accurately and fast. You Only Look Once (Yolo) is one of the most popular algorithms for object segmentation and detection. Yolo is a cutting-edge, real-time method for solving object recognition problems based on deep learning. The algorithm divides the image into specified bounding boxes, then runs a recognition algorithm in parallel for each of them to determine which object class they belong to, before intelligently combining the boxes to form optimal bounding boxes around the objects. At test time, YOLO examines the entire image, so its projections are influenced by the image's overall context. Previous methods, like Region based convolutional neural network (R-CNN) and its various forms was using a pipeline to complete this

process in several stages. Since each part must be trained independently, this can be sluggish to run and difficult to refine. Yolo accomplishes all of this with a single neural network.

Convolutional Neural Networks (CNN), such as Yolo-V3 and Yolo-V4, are Deep Learning methods used to construct the machine. When the performance of those models are compared.

The most common deep-learning frameworks and libraries, such as Tensorflow, Keras, and OpenCV, are used to create a CNN-like deep learning model. They're commonly used in real-time apps and applications. The proposed model is also being used to develop end-to-end software that scans each person before they attend any public meeting.

In this work, I am proposing the idea of designing the document image classification system using Yolo that could help to identify the document's physical layout and figure out the segment portions in a particular document.

1.2 Scope for this project:

The aim of this research is to boost classification accuracy and performance of the system and most importantly, to use it to classify the segmentation of the document to help the features extraction accurately. To achieve this, various key questions have been identified, which help to solve the problem and help to achieve the objectives.

1.2.1 Aim

As it is hard to accurately extract features from a trove of documents manually. Automatic segmentation and classifications of the document are necessary to extract the document features in more reliable and effective ways.

The proposed models used in this study consist of two deep-learning model comparisons. Both are using transfer learning approaches. Transfer learning is a methodology in which a CNN model generated on one task is used on the foundation of another model. Given the significant computational and time needed to construct a model for these problems, as well as the significant improvements inability that they have in relating cases, it is a common practice in deep-learning to use pre-trained models as a basic foundation for natural language processing and compute vision tasks [10]. Transfer learning used in this work is Yolo-V3

and Yolo-V4 to identify document segments such as author, title, the body of text, figures, tables, page number, display equation, header, reference, caption, and footer. The primary goal of our research is to develop an integrated method for document segmentation and classifications. This system will help automatic detection of each document with details of portions on each page. It will help to understand a document more reliable and easy.

1.2.2 Objective

A big challenge in document analysis is the difficulty of accurately segmenting and classifying the region of the document from formatted PDFs. We need to develop a system that automatically classifies the segments of a particular document in the given input. Automatic document segmentation is one of the best ways to find out the author's contributions and materials such as title, figures, tables, etc in a document.

A deep-learning-based model for automated document segmentation and classification. Both of these things help the network's success, reliability, and accuracy. The following is one of the study's main goals: Since such a CNN architecture is needed, our goal is to train the network to achieve improved efficiency and accuracy by using fewer parameters.

1.3 Problem Statement

Building a (Computer-Aided System) CAS based system requires an automated document segmentation and classification system [56]. The lack of adequate training data is a big issue in the field of automated document segment detection [11]. The cost of image collection is high due to two reasons [11]. The method of gathering the collections is very complex, which is one of the major issues. The labeling or naming of the dataset, on the other hand, necessitates a significant amount of time and effort.

There are some machine learning methods for detecting and classifying text segments; however, the issue with those strategies is that they depend heavily on the correct descriptor selection and do not perform end-to-end classification and detection. DocBank dataset provides large-scale dataset using weak supervision approach, however, the bounding boxes and types of labelling are not suited for this work. self-labelling is recommended to use this dataset [36]. To tackle

the problem of the limited numbers of dataset and automatic detection and segmentation, deep-learning pre-trained models are recommended. Recently, models like Region based convolutional neural network (R-CNN), Faster-RCNN, Residual Network (ResNet), and many others are the current state-of-art. These models solve these issues and deliver outstanding results in a variety of fields. However, losing information to the vanishing gradient problem with VGG and AlexNet [63], some detail is lost in subsequent layers. ResNet is proposed as a solution to this issue. To merge previous details within the next layers, ResNet employs skip links. While, for R-CNN and its variants, to execute this process in multiple stages, is using a pipeline. Since each part must be trained independently, this may be slow to operate and difficult to refine.

This pre-trained models, on the other hand, use transfer learning methods, and the issue with those networks is that their architecture nature requires a lot of parameters. As a result, it takes a long time to train the models. The CNN model, on the other hand, will automatically identify segments and label images [34].

1.4 Deliverance

1. A deep learning model for document segmentation and classification.

2. Dataset collection, data pre-processing, and network selection are all steps in the process. Python, OpenCV, TensorFlow, Keras, Pandas, Numpy, and Sklearn are examples of deep learning software.

3. Existing work, Dataset Descriptions, Methodology, Experimental Setups, and Results are all included in the final study.

5. The original dataset of document segmentation is available in DocBank dataset at https://doc-analysis.github.io/docbank-page/.

Chapter 2

Literature Review and Existing Methods

The aim of this chapter is to go over all of the methods that were used in the text segmentation and classification. A solid foundation for fully comprehending the activities in this study has been created. This section also covers document segmentation and grouping, as well as many of the related topics. Different principles, methods, models, and deep learning have been developed to improve identification and classification.

2.1 Related Work

Several kinds of research have been done for document segmentation and classification. Related research of document segmentation and classification is mainly using machine learning, specifically: deep-learning techniques. Many different datasets for documents are available for segmentation and classification tasks, however, most of them are not labeled. Even if some datasets are labeled, they are not suited for the proposed models. So, self-labeling is suggested for this project.

Face detection for identity authentication and image restoration was the subject of many research projects. However, the main objective of this study is to categorize the documents section such as authors, title, figure, etc.

Oliviera et al [50] proposed a CNN-based technique for the segmentation of documents. The proposed technique is based on the CNN-based classifier partnered, having task-dependent on post-processing boxes. On the other hand, Augusto et al [**?**] suggested a fast CNN-based architecture. As opposed to a conventional bi-dimensional CNN solution, the author claims that the proposed model can achieve quicker execution times and more compact data use without sacrificing overall precision.

Long et al [25] used an automatic feature extraction and classification with Fully Convolutional Network (FCN) for semantic segmentation. FCN is a convolutional network data layers that have a three-dimensional sequence of h*w*d size for each layer, with The spatial dimensions are h and w, and the function or channel dimension is d. The image is on the first sheet, with h w pixel sizes and d color channels. Higher-layer positions correspond to their receptive fields, which are the image locations to which they have a direction [38].

In Deng et al. [24] paper the author suggested a pre-trained network based on the database of the imageNet. The network has up sampling de-convolutional layers and marge the top prediction layers of the network with skip connection to improve the performance.

The author in [51] proposed that FCN's architecture is extended by making the expansive path's (decoder) parameters symmetric to the contraction path's encoder. The proposed network looks like a U-shape network with some skipping connections for every steps. Similarly, Convolutional Neural Networks (CNNs) have widely adopted and proven to show remarkable results in many fields such as VGG-16 [29], Resnet [20] and Alexnet [1] architectures. To some extent, many researchers have experimented with using CNN for document analysis. Finally, some research papers that are worth mentioning for document processing and detection task are: [15] , [42] and [23].

Several methods have also been adopted with neural network architectures, such as U –shaped CNN network and MD-LSTM model for pixel-wise segmentation tasks [27],[66] and [64].

2.2 Existing Methods

Different deep learning tools and strategies, as well as their components and measurement matrix, are discussed in this part of the study.

2.2.1 K-nearest Neighbors (K-NN)

The supervised machine learning algorithm k-nearest neighbors (KNN) is an algorithm that is easy and simple to implement that can solve both regression and classification problems [19]. KNN is a classification model that classifies data points depending on how close they are to each other. It makes an "informed guess" on what an unclassified point can be classified as based on test results [58].

KNN calculates the distance between the dataset's instances and creates centroids dependent on these distances, which are then labeled as groups [8]. Calculating the difference between a sample and all of the examples in the data, selecting the K closest examples to the query, and either voting for the most regular label (in the case of classification) or averaging the labels are how KNN functions (in the case of regression). Experimenting with a few different Ks and choosing the one that suits best is how to choose the best K for data.

$$d(p,p) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \tag{2.1}$$

2.2.2 Deep Learning Models for Documents classification

Deep Learning layers virtual neurons to build complex artificial neural network architectures. It is the task of each neuron to add up all of the inputs and determine whether or not to give an output signal to the layer of neurons above it. Any neuron in a layer in the network is connected to neurons in the layers above and below it. By learning the best weights for each of these connections, this neural network, like our own brain, can solve a wide variety of problems. Even though a neural network is a simple term, the sheer amount of connections between neurons enables it to represent extremely complex problems [26].

Although deep learning is a form of machine learning, it has gotten a lot of attention because of its adaptability – it is based on how our own brains function. It is often thought to be superior because it offers a complete approach to classification issues. It is a superior end-to-end approach since it can remove functionality using CNN-based architecture. For attribute classification and extraction of Histopathological photographs, various Deep learning models were used. Just a few of them are covered in this article.

2.2.2.1 Convolutional Neural Network

In order to solve image recognition issues, CNN models are now called state-of-the-art models. Many layers, such as input-output and some unknown layers for sorting, make up CNN templates. Convolutional layer, pooling layer, and dense layer are the three primary layers of this model (Fully connected layer).

Convolution Layer: A convolutional layer is made up of a number of filters, each of which has its own set of parameters that must be studied. The filters' height and weight are less than those of the input length. Each filter is convolved with the input volume to produce a neuron-based activation map [28]. This layer contains a series of kernels that slide over the image in the width and height directions, producing a 2d activation map for each kernel. These activation functions, such as RELU, Sigmoid, Tanh, and SoftMax, are used to apply non-linearity to activation maps.

Pooling layer: In most cases, pooling layer is incorporated between convolutional layers of two. By down-sampling the representation, the pooling layer reduces the number of parameters and computation [28]. Values are derived from the activation maps using stride and pooling techniques. There are functions for

pooling, such as Max pooling and Average pooling. Mostly, Max pooling is used rather than Average pooling because it works better.

Layers that are completely connected are also known as thick layers or flatten layers. Features of convolution layers derived from previously hidden layers are flattened into a 1-D array in these layers for performance classification [12].

The CNN model are shown in Fig 2.1.

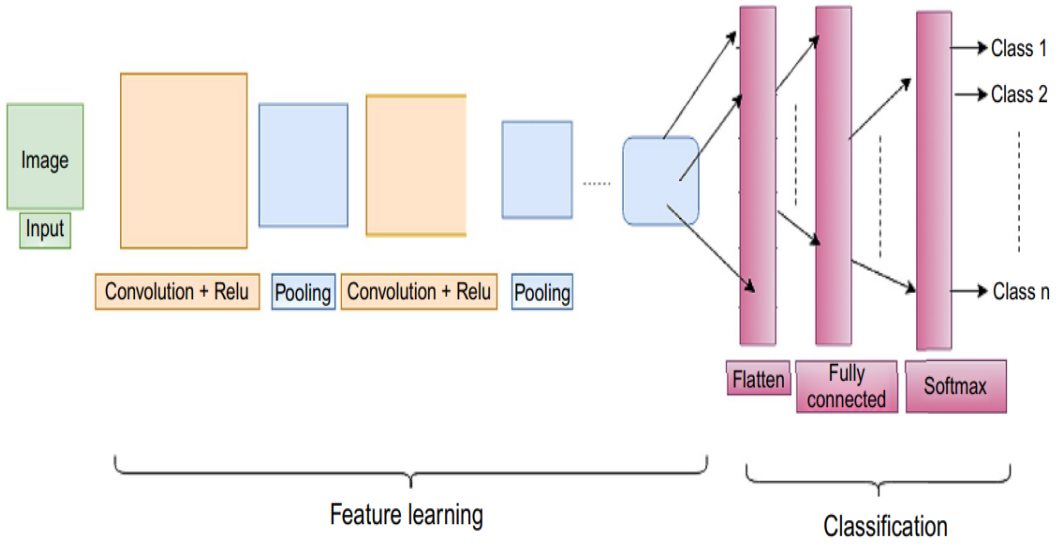

FIGURE 2.1: Simple Convolution neural network

Object detection is another important aspect of computer vision. Pose estimation, vehicle detection, and surveillance all benefit from object detection. Object detection algorithms differ from classification algorithms in that they aim to find the object of interest inside the image by drawing a bounding box around it. Furthermore, in an object detection problem, there should be more than one bounding box [16].

The main reason why this issue can not be solve by using a regular convolutional network with a completely connected layer is because the output layer length of is not constant, it is in variable, the number of instances of the detected object isn't fixed. One simple solution to this should be removing the various interested regions from the image and use a CNN to detect within each region of the object. The difficulty of this technique is that the points of significance inside the image will be in different spatial locations and have different aspect ratios. As

a consequence, a significant number of areas must be chosen and used, which may take a long time [17].

2.2.2.2 Region Based Convolutional Neural Networks

The R-CNN method attempts to solve the problem of locating objects in an image (object detection). To address this issue, a sliding window method can be used. When using this approach, though, it simply went through the whole image with various rectangular sizes and using the brute-force method, to look for certain smaller images. The issue is that there will be a massive amount of smaller pictures to look at [13].

To deal with this problem, Ross Girshick et al [17]. suggested a way to avoid this issue of choosing a large number of regions. He named them region proposals because selective search is used to remove only 2000 regions from the image. As a result, rather than having to identify a large number of regions, it will now deal for only 2000 regions. For the next step, a CNN can be used to create a function vector for each area proposal that represents this picture in a much smaller dimension. Finally, a Support Vector Machine (SVM) classifier, which analyzes data for regression analysis and classification will be used to classify each feature vector for any object class. This last step is uses for classifying the regions of the image.

R-CNN model are shown Fig 2.2.

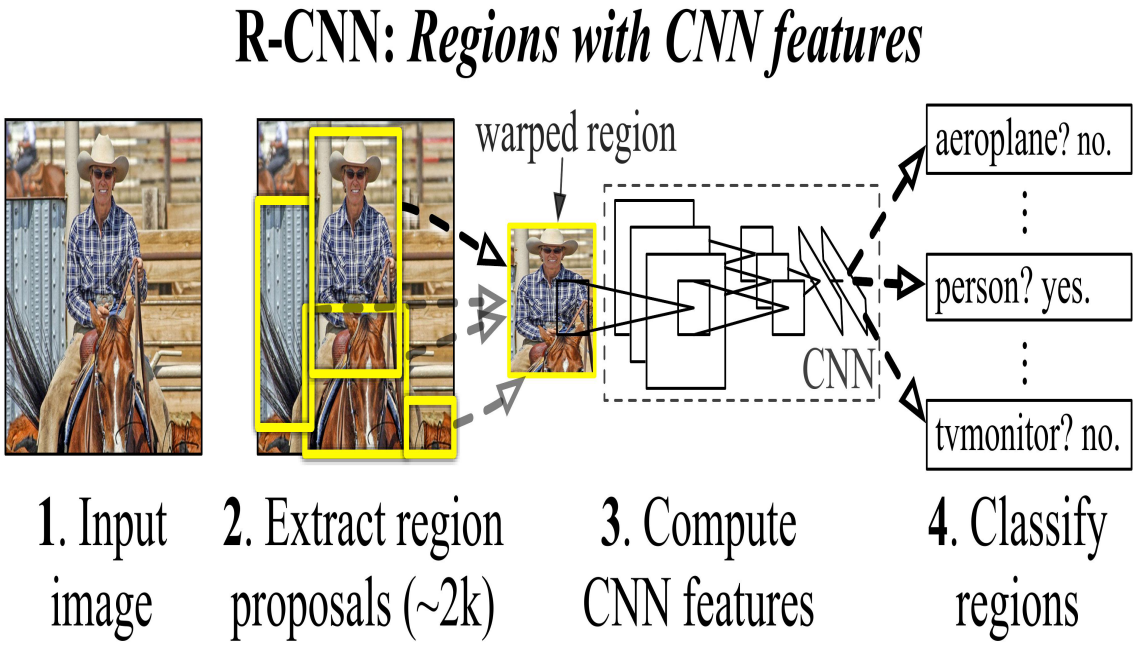

FIGURE 2.2: R-CNN: Regions with CNN features. Towards data science, by Rohith. Gandhi, 2018, https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e

2.2.2.3 VGG-16

In 2014, VGG-16 was one of the first CNN-based models proposed. The first ImageNet competition was won by this architecture. They changed the previously proposed Alexnet architecture by changing the kernel size filter from 11 to 5 of size 3x3 in the kernel size filter. They have 13 convolutional layers and three thick layers in their structure [29]. This architecture is shown in Fig 2.3.

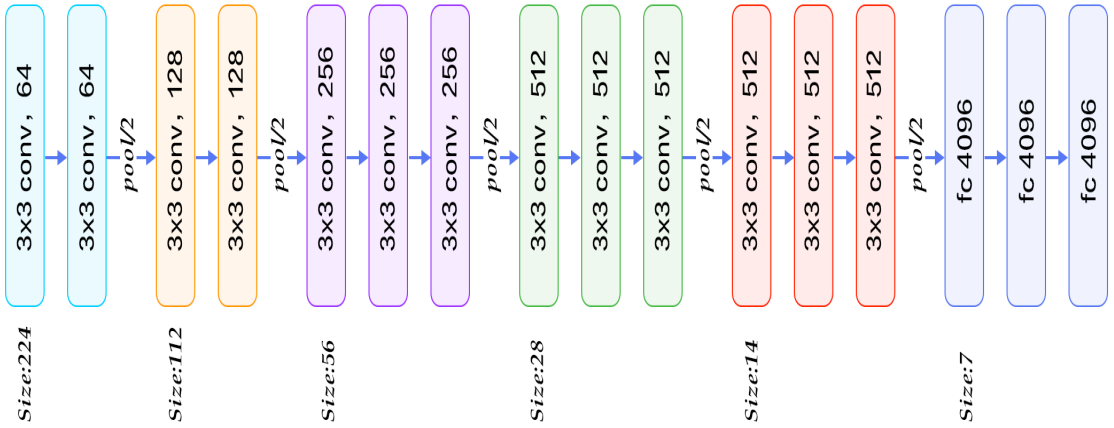

FIGURE 2.3: Image Classification using pre-trained VGG-16 model, by Roshan, 2020, https://kgptalkie.com/image-classification-using-pre-trained-vgg-16-model/

2.2.2.4 Residual Neural Network

The residual relation is used in the ResNet residual network model. This is one of the most widely used models for problems including sorting, segmentation, and target identification. In 2015, this model won the ImageNet competition. Before ResNet model, the deep learning model has an issue of unstable behaviour that it may run into when training a deep neural network, which is a vanishing gradient problem.

Resnet model is shown in Fig 2.4.

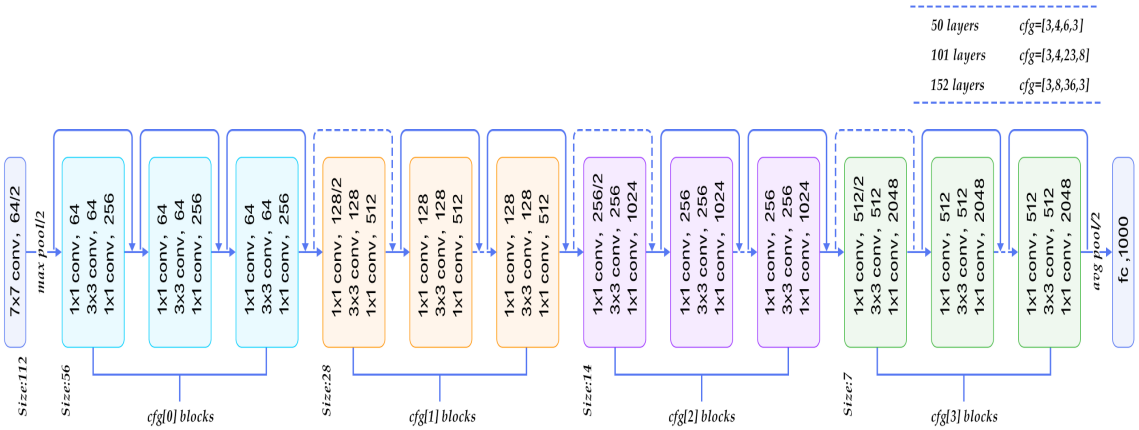

FIGURE 2.4: CNN Architectures, a Deep-dive, by S. K. Basaveswara, 2019, https://towardsdatascience.com/cnn-architectures-a-deep-dive-a99441d18049

2.2.3 Deep Learning Tools for Image Classification

With the emergence of deep-learning-based architectures, a slew of deep-learning-based frameworks have emerged to aid developers in coding their ideas for segmentation, classification, and feature extraction. Although the mentioned methods and techniques above are discussed, to be able to understand this better, the following tools are described:

2.2.4 Evaluation of Documents Classifications

The accuracy of data segmentation and classification images is evaluated using the following matrices in general: Precision, recall, F1-score, Macro average, and MCC metric are all determined from the uncertainty matrix [21]. The confusion matrix is usually made up of True Positive (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), as seen in table 2.1, with the equations below.

TABLE 2.1: Confusion Matrix untuk Evaluasi Model pada Supervised Learning, by K.S. Nugroho, 2019, https://medium.com/@ksnugroho/confusion-matrix-untuk-evaluasi-model-pada-unsupervised-machine-learning-bc4b1ae9ae3f

		Actual	
		Positive	**Negative**
Predicted	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.2}$$

$$Precision = \frac{TP}{TP + FP} \tag{2.3}$$

$$Recall = \frac{TP}{TP + FN} \tag{2.4}$$

$$F1 - Score = \frac{2 \times Precision\ xRecall}{Precision + Recall} \tag{2.5}$$

$$Macro - F = \frac{1}{a} \sum_{k=1}^{a} FK \tag{2.6}$$

Chapter 3

Proposed Methodology

This chapter's aim is to demonstrate that the proposed model has resolved the shortcomings of previous surrogate models. The study issue has been answered as well. This paper discusses the suggested methods for record segmentation and classification.

3.1 Proposed Methodology

I have proposed different deep-learning models and also discussed them in the previous chapter. Looking at the issues the previous models have dealt with, in this work, the suggested models that can improve from before and also best suited for document segmentation and classification are Yolo-V3 and Yolo-V4.

The trained models are Yolo-V3 and Yolo-V4 on Google Colab. To train the models, transfer learning is done to these models with Darknet. The darknet model will be converted to Tensorflow. The performance of both Yolo-V3 and Yolo-V4 will be discussed in the next chapter.

The system model diagram is shown in Fig 3.1

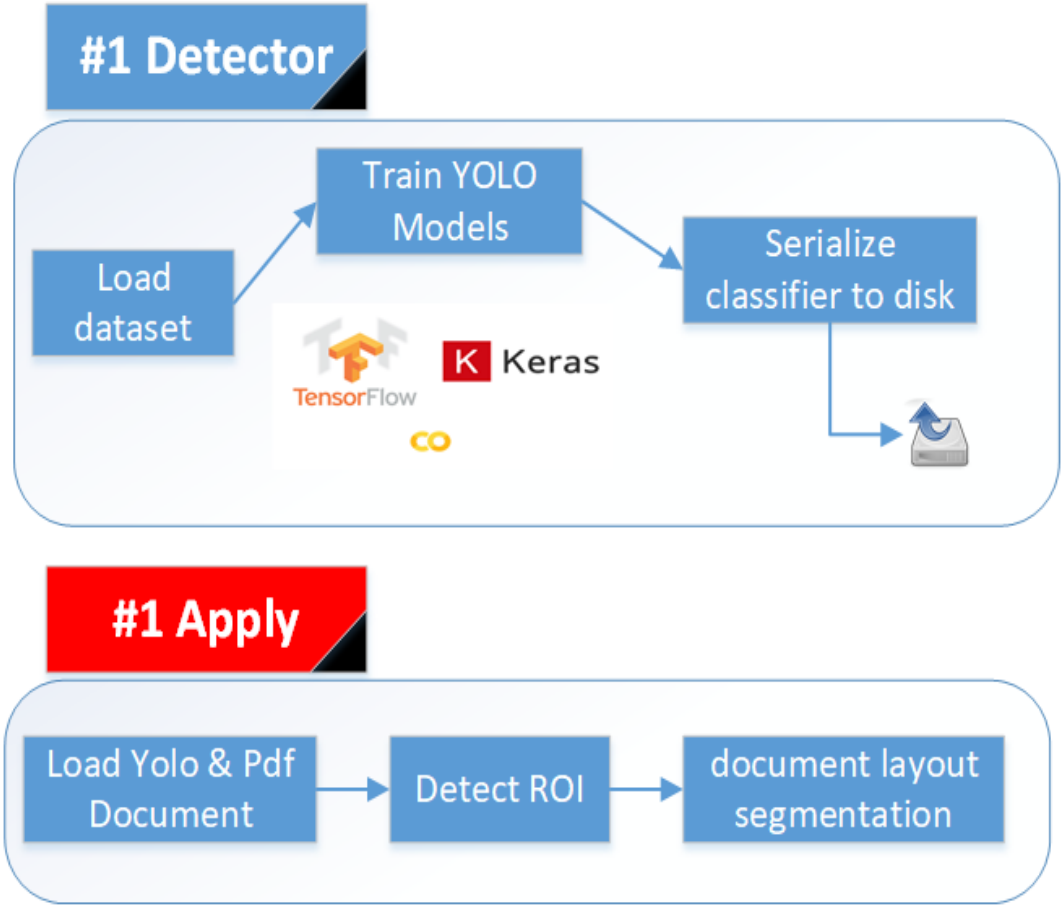

FIGURE 3.1: System diagram of proposed approach's

In this proposed system, The way the system work is that, at first, the pre-trained model of the darknet is called upon on Google Colab. Then deploying the labeled training dataset onto the local folder on the darknet. After deploying them on the darknet folder, the models are then trained by using Keras and Tensor-Flow. After Training the models, download those classifiers or models into the local PC. In the last step, load those models again to predict the images.

However, as we all know that document mostly comes in a pdf format, not images, the system is made to enables the user to submit a pdf file. The pdf file that is submitted will then be converted into images for each page by the system so that the prediction can be made.

3.1.1 Darknet

Darknet is a neural network system that is free and open source. It is a quick and highly accurate system for real-time object detection (accuracy for custom trained models is dependent on training data, epochs, batch size, and other factors) . Also can be used for images. The fact that it is fast and accurate is because it is written in C and uses CUDA [69].

3.1.2 Transfer Learning (TL)

Transfer learning (TL) is a technique to use a model developed for one task as the foundation for another task's model. [10].Deep learning, on the other hand, is the method of automatically extracting and classifying features [14]. In the sense of deep learning, a deep convolution neural network needs a large volume of training data. Where the necessary dataset volume is insufficient, deep CNN pre-trained models are used. Since these pre-trained networks have already been trained on a vast number of videos, sharing their information is an act of TL. Following the transition of information, CNN's pre-trained network is fine-tuned using a limited volume of new data. TL methods are often used in deep-learning applications because they are quicker and simpler to implement than creating a model train from scratch with random weights. Low-level features (information) are extracted from initial layers including curves, colors, edges, and corners in CNN models. It goes far, with more abstract, essential (meaningful) features (information) extracted at the final layers [1]. Pre-trained models are trained on ImageNet datasets instead of medical datasets in real-world applications like medical imaging, so fine-tuning is needed for better classification. Top layers or totally linked layers (Dense layers) are modified and frozen in a pre-trained

network. Softmax is withdrawn from the grouping layer and replaced by new ones. And, for classifications, the fine-tuned model is used. According to recent studies, TL is a more accurate and useful method for the classification of BC tissues in histopathology photographs, as seen in Fig.3.2.

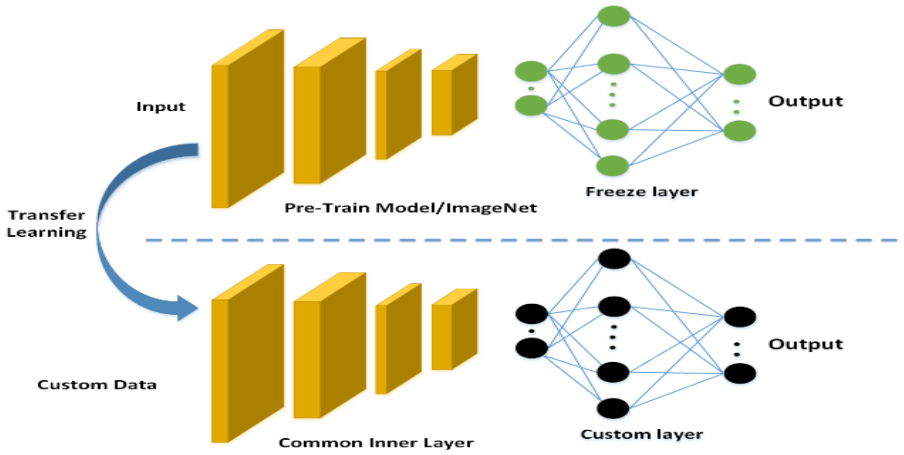

FIGURE 3.2: Transfer learning (TL): top layers, which is the last layers of the models are fine-tuned

3.2 CNN Architectures

For automated feature extraction and classification of images, the CNN has several layers, including convolution layers, pooling layers, and thick layers (fully linked layers). CNN has proposed a number of deep learning architectures for classification; the most common shallow CNN networks are VGG and AlextNet. The vanishing gradient problem is a weakness of these two networks. Modern CNNs, such as the most commonly used networks ResNet and DenseNet, have complex connections among the layers to mitigate the vanishing gradient problem. These networks faced a variety of problems, including feature map redundancy, single-dimension feature scaling (such as depthwise or widthwise feature scaling), and network saturation. Furthermore, since these architectures had a vast number of parameters in the network, training took a long time and needed a lot of memory.

As we all know, machine learning methods are heavily influenced by the descriptor chosen for feature selection. However, the ability of the deep-learning model is the end-to-end networks [35]. They could perform automatic segmentation and classification. Also compare to previous approaches, deep-learning models have produced remarkable results.

3.3 YOLO (You only look once)

The R-CNN family methods rely heavily on regions to locate objects in images. The network only looks at sections of the picture that have a better probability of possessing an object, rather than the whole image [59].

Item detection is handled differently in the YOLO system (You Only Look Once). Using the entire picture as a single instance, it calculates the bounding box regions and the probabilities of the classes for these boxes. The greatest achievement from YOLO is the amazing speed; it can deal with 45 frames for each second. Object representation is also something that YOLO is mindful of [54].

This is one of the better object detection algorithms, with an efficiency that is comparable to the R-CNN algorithms [59].

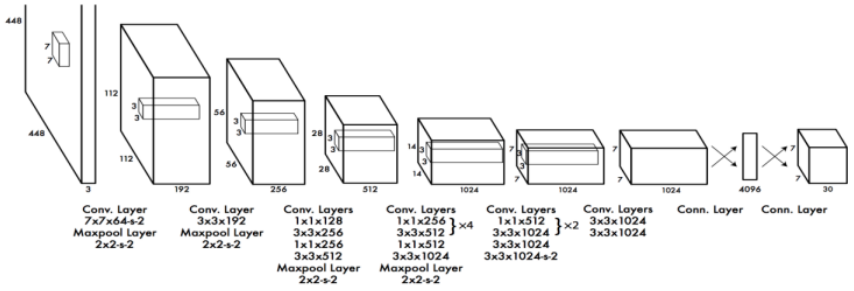

FIGURE 3.3: Complexity and accuracy analysis of common artificial neural networks on pedestrian detection, by Jiatu. Wu., 2018, https://www.researchgate.net/publication/329038564

As seen in Fig 3.3, The algorithm divides the image into bounding boxes, then runs a recognition algorithm for each of them in parallel to decide which object class they belong to, and then intelligently merges these boxes to form optimal bounding boxes around the objects [39].

R-CNN and its families, for example, was using a pipeline to execute this process in several stages. Since each part must be trained independently, this can be sluggish to run and difficult to refine. YOLO accomplishes all of this with a single neural network [39].

3.4 Yolo-V3

3.4.1 Bounding Boxes

By doing logistic regression, the score on each bounding box is predicted by Yolo-V3. The value is 1 if the anchor box does overlapping the ground truth box rather than the other anchor boxes. Anchor boxes that overlapping but are not as solid as the land truth box would be ignored. They had set a 0.5 as a minimum threshold. As a result, the bounding box can only have one anchor box. To locate the anchor boxes, k-means clustering is used [46]. K-means clustering is sort of unsupervised learning. It is used when dealing with unlabeled data (i.e., data with no categories or groups that are defined). The aim of this algorithm is to locate the classes of data, with K representing the instances of the classes. Based on the features given, the algorithm assigns each data point to one of K groups iteratively [67].

3.4.2 Prediction

Yolo-V3 uses binary cross-entropy loss and independent logistic classifiers for class prediction during the training session. Complex datasets can now be used for Yolo-V3 model training thanks to these changes [43].

Yolo-V3 uses a multi-label approach, which allows for more complex classes and multiple bounding boxes. Yolo's previous implementation, on the other hand, relied on a mathematical equation that converts numbers of a vector into the probabilities vector, called softmax, with each value proportional to the probabilities of the vector's size. Every bounding box can only belong to a class by using the softmax, which is not always the case.

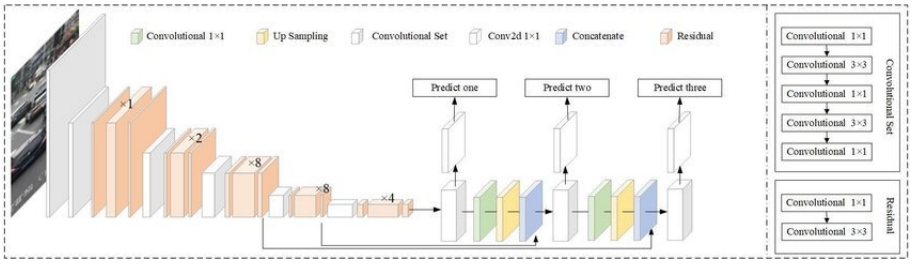

FIGURE 3.4: Mini-YOLOv3: Real-Time Object Detector for Embedded Applications, by R. S. Jia, 2019, https://www.researchgate.net/publication/335865923

3.5 Yolo-V4

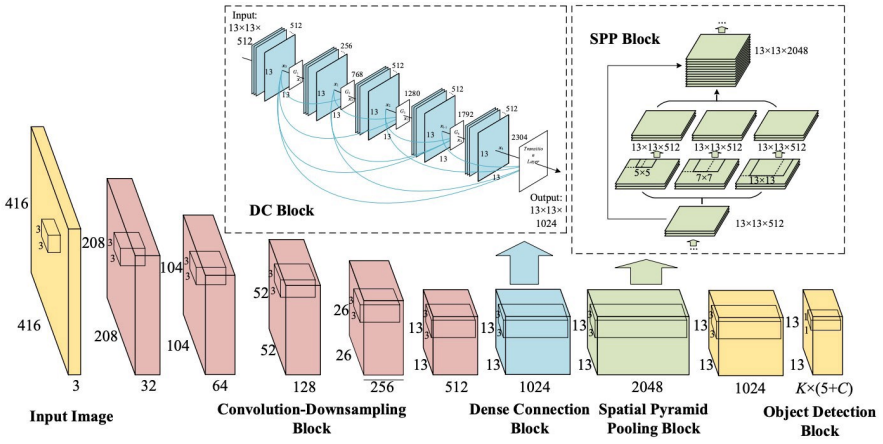

FIGURE 3.5: YOLOv4 — Version 3: Proposed Workflow, by Deval Shah, 2020, https://medium.com/visionwizard/yolov4-version-3-proposed-workflow-e4fa175b902

Yolo-V4 outperforms current methods in terms of accurate detection and the superior speed of training time [68]. According to Bochkovskiy, A. et al [7], the research team describes it as a "speedily working" object detector that can be easily trained and used in production systems.

Yolo-V4 incorporates the latest BoF (bag of freebies) and other BoS (bag of specials). The BoF improves the detector's accuracy without increasing inference time. They merely raise the expense of teaching. The BoS, on the other hand, increases the inference cost by a slight amount while greatly improving target detection accuracy [60].

3.5.1 Backbone

An object detector's backbone network is typically pre-trained on ImageNet classification. The network's weights have already been optimized to classify appropriate features in an image, but they will be tweaked for the new task of object detection [61]. There are three components of the Yolo-V4 backbone architecture, which are: CSPDarknet53, Bag of freebies, and Bag of specials.

3.5.1.1 CSPDarknet53

On top of Darknet53, there is a CSPNet (Cross Stage Partial Network) network that is called CSPDarknet53. Darknet53 employs the ResNet principle to make

surer the network has depth, while tackling the vanishing gradient problem. The CSPNet will improve CNN's learning ability while lowering the computation and memory costs [18].

3.5.1.2 Bag of Freebies (BoF)

Methods that fall into the bag of freebies category either increase the cost of training or change the training strategy while keeping the cost of inference minimal. There are a few basic methods for this that are widely used in computer vision, such as: Data Augmentation, Focal Loss, Label Smoothing, and IoU Loss [57].

3.5.1.3 Bag of Specials (BoS)

The bag of special methods is a set of methods that raise inference costs slightly while vastly enhancing target detection precision and accuracy. One example of methods for BoS is Mish activation [57].

3.5.2 Neck

Between the backbone and the head, there are extra layers. They're used to remove various feature maps from various backbone stages. The neck element may be an Feature Pyramid Networks (FPN), Path Aggregation Network (PANet), or Bi-directional Feature Pyramid Networks (Bi-FPN), for example [3]. Instead of using Yolo-V3's FPN for object detection, PANet is used for parameter aggregation for various detector levels in Yolo-V4. PANet was chosen for instance segmentation in Yolo-V4 because of its ability to accurately retain spatial information, which aids in proper pixel localization for mask formation [44].

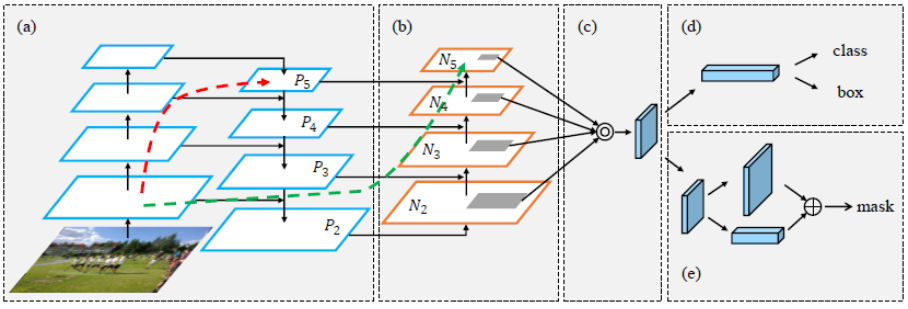

FIGURE 3.6: PANet: Path Aggregation Network In YOLOv4, by Miracle. R., 2020, https://medium.com/clique-org/panet-path-aggregation-network-in-yolov4-b1a6dd09d158

3.5.3 Head (Detector)

This network is in charge of detecting bounding boxes (classification and regression). Depending on the implementation, a single performance might look like this: its likelihood of k-classes + 1 and four values of the bounding boxes (x, y, h, w) [3]. Infact, it is still the same detector as Yolo-V3.

3.6 Working with the model

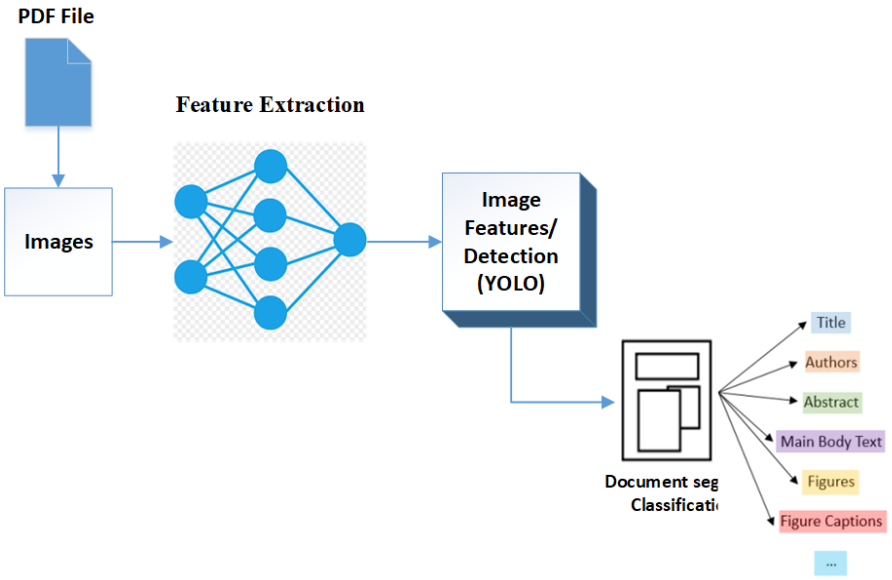

FIGURE 3.7: Model Diagram

Using Yolo-V3 and Yolo-V4 models as the networks for document segmentation and classification. The networks are fine-tuned, passed our custom data. The proposed methodology takes an input of 416 × 416 and applies pre-processing steps and normalization as seen in Fig 3.7. The images is fed into the proposed model for finding the region of interest and then evaluate the performance of the network.

3.7 Dataset characteristics

The Dataset for this project is taken from Docbank Dataset. The dataset is publicly available at https://doc-analysis.github.io/docbank-page/. It contains 500K document pages [36]. However, the labeling of the bounding boxes and classifications are not best suited for this project. Hence, self-labeling is done by

only using 500 document pages as the training data and 50 document pages as the testing data.

For this work, I used 11 classes for labeling the dataset, which is: author, title, header, footer, pageNumber, bodyOfText, figure, table, caption, reference, and displayEquation.

3.8 System model

To develop the system model, first of all, was collecting the dataset. Then, Labeled the dataset depending on needs. Next, was training the labeled training data to Yolo models using Tensorflow and Keras. After downloading the models and necessary files into the local pc, a pdf converter will be developed on the system for enabling pdf as input and converting the pdf into images of each page. Then, The developed system can do the segmentation and prediction while the converter converts the pdf into images as seen in Fig. 3.7.

Fig. 3.8 shows the process of how the system works in a flow chart diagram.

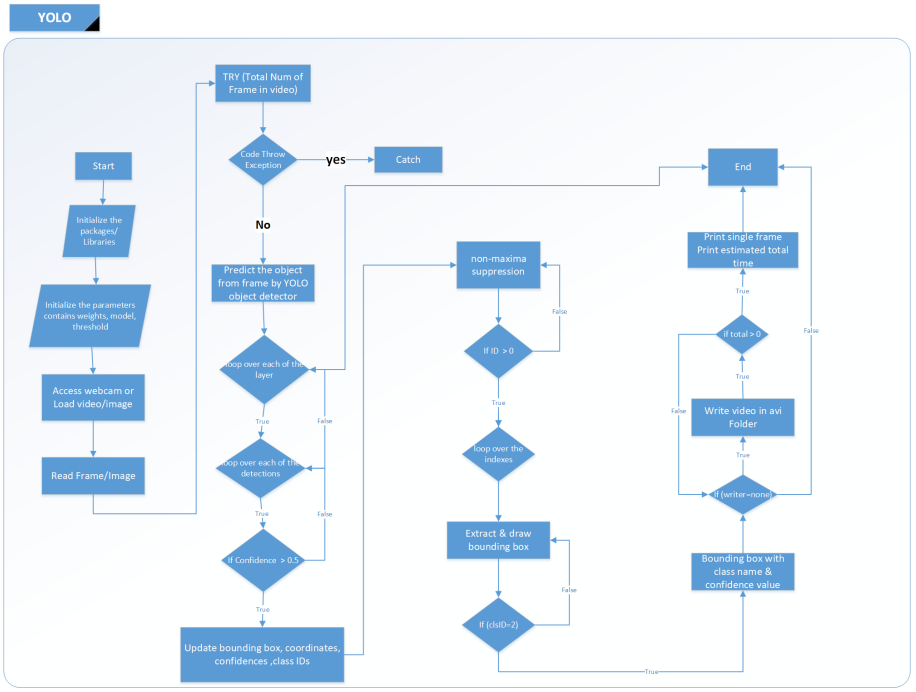

FIGURE 3.8: Flow Chat Diagram

Chapter 4

Evaluation and Experimental Results

The objective of this chapter is to evaluate the loss function of Yolo and performed experiments on document segmentation and classification to see if the proposed methodology is working.

4.1 The Loss

YOLO has a very small and straightforward topology. The fact that its loss function is very complicated allows it to have such a simple structure, or more precisely: compact. The absence is what gives the features their significance. As a result, a well-thought-out loss function will pack a large amount of data into a small feature map [2].

To measure loss, YOLO is using the sum-squared error between the ground truth and the predictions [22]. The loss function is made up of the following components:

1 the coordinate loss - a box prediction that does not completely cover an item.

2 the objectness loss - an incorrect box-object IoU prediction.

3 the classification loss - due to erroneous predictions of '1' for proper classes and '0' for all other classes for the object in that box.

Final loss adds coordinate loss, objectness loss, and the classification loss together.

$$
\begin{aligned}
&\lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2 \right] \\
&+ \lambda_{\textbf{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i} \right)^2 \right] \\
&+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i \right)^2 \\
&+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i \right)^2 \\
&+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} \left(p_i(c) - \hat{p}_i(c) \right)^2
\end{aligned}
$$

FIGURE 4.1: Loss Function of Yolo

4.2 Experimental Setup

Following are the setup to perform this experiments:

1 Experiments performed on Google Colab platform with GPU of NVIDIA (R) Cuda compiler driver.

2 Deploying the darknet pre-train Yolo models on Goole Colab.

3 Then, upload the labeled training data to the darknet folder.

4 Train the the models with the labeled training data.

5 Download the necessary files to local folder.

6 Develop the system with the chosen Yolo models to enable converting pdf to images (JPEG or PNG format).

4.3 Experiments

Here will discuss the testing data that have been experimented with from DocBank Dataset. The training data used in this project is 500 images, while the testing data used in this project is 50 images. The performances are manually evaluated using the testing data that is converted from pdf to images. The accuracies that I took into account are categorized into two parts: the accuracy of correctly predicting the bounding boxes and the accuracy of whether the bounding boxes missed any regions of interest.

4.3.1 Document Segmentation and Classification Results:

4.3.1.1 Yolo-V3 Results

Table 4.1: Yolo-V3 results

Yolo-V3	**Acc.**
Accuracy of the predicted regions	92%
Accuracy of the segmented areas	83%

It took 0.45 seconds average for Yolo-V3 model to predict 50 testing data with the accuracy of 92% (46 out of 50) for predicting the correct class for the bounding boxes and 83% (40 out of 50) for placing the correct bounding boxes. Noticing the performance of segmenting the interest's regions are not high, also noticed that when trained lesser training data of that class will end-up with low probability of it being the correct classes. For example, classes like figure, table, title and author, Trained Yolo-V3 model used in this work is showing not satisfying results for predicting those classes.

Here are some of the experimental results of Yolo-V3 on:

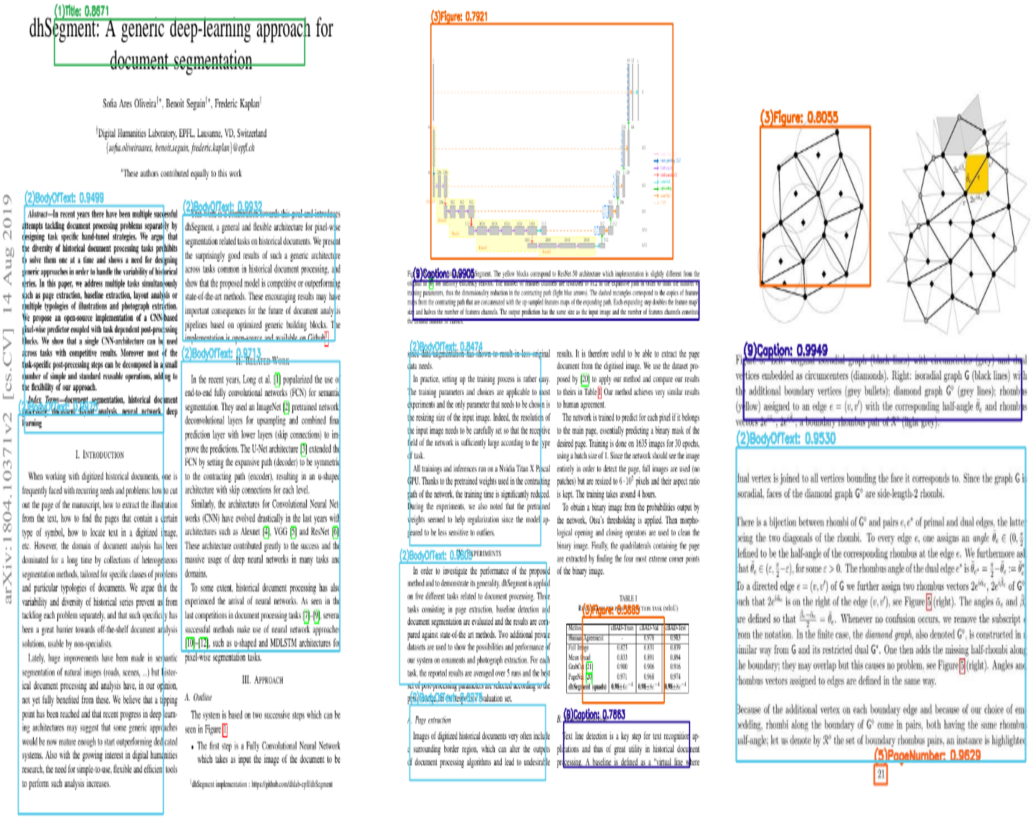

FIGURE 4.2: Yolo-V3 Prediction Results

4.3.1.2 Yolo-V4 Results

TABLE 4.2: Yolo-V4 results

Yolo-V4	**Acc.**
Accuracy of the predicted regions	98%
Accuracy of the segmented areas	92%

It took 0.20 seconds average for Yolo-V4 model to predict 50 testing data with the accuracy of 98% (49 out of 50) for predicting the correct class for the bounding boxes and 92% (46 out of 50) for placing the correct bounding boxes. Noticing the performance significantly improve compared to Yolo-V3 model on both segmentation and classification. In this model, prediction of the classes is mostly accurate on all of the testing data, only one-out-of fifty was showing incorrect prediction, which is the class: author was predicted wrong.

Here are some of the experimental results of Yolo-V4 on:

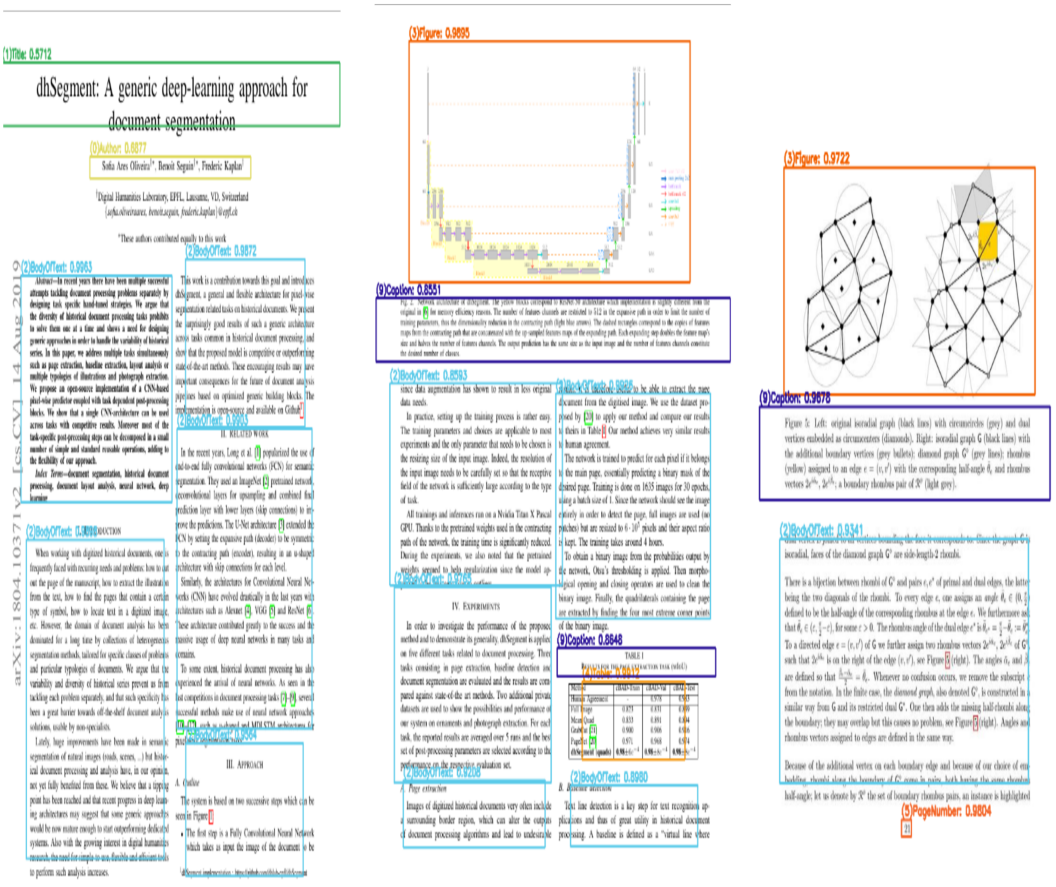

FIGURE 4.3: Yolo-V4 Prediction Results

4.3.1.3 Comparing the results

Here is the model comparison of Yolo-V3 and Yolo-V4 table 4.3. The results shown that Yolo-V4 give better accuracy as compare to Yolo-V3, which is expected.

TABLE 4.3: Model Results Comparisons In Term of Accuracy

Model	Acc. of bounding boxes	Acc. of the predictions	Speed of predicitons
Yolo-V3	83%	92%	0.45 seconds
Yolo-V4	92%	98%	0.20 seconds

The huge gap on the accuracy of the bounding boxes is expected, as Yolo-V3 is still using Feature Pyramid Networks (FPN), while Yolo-V4 is using Path Aggregation Network (PANet) for segmentation detactor.

Chapter 5

Conclusion

5.1 Conclusion

This works offer a state-of-the-art detector which is faster and more accurate in performance compared to other recent methods. The detector described can be trained and used on other detection and classification cases, this makes its broad use possible. The concept of Yolo models for segmentation and classification of a document has proven its viability. A large number of methods have been verified, and selected the best one for use to improving the accuracy of both the classifier and the detector. These features can be used as a model for future research and development.

5.1.1 Make it Better

Time constrain is the biggest challenge while doing this project. Normally, for pre-trained CNN model like Yolo, It is suggested that each label have about 1000 annotations. In general, the greater the number of images per mark, the better your model will work. Increasing the current training data, which is 500 training dataset would significantly improve the accuracy of the bounding boxes placement. One should also mindful when labeling the dataset for training the model. It could have been easier if the labeling for body of text class change into paragraph class, and labeled each paragraph instead of the whole paragraphs in the body of text as one entity.

5.1.2 Future Work

As mentioned in the above section, increasing the training data and choosing better class for labeling would make the model performance better on classification and segmentation. Improving the model into such extends that it is possible to analyse historical document or could also work with language that use ideogram (for example: Chinese and Japanese) or diacritic marks (Arabic and Hebrew) or even handwritten document.

Developing the system and model to such extends that after the segmentation and classification is done, feature extraction is also made possible by using some of the CNN methods. Extracting features from the document into their specific format files would make literature mining and document analysis easier. With the help of some different cognitive processes such as Optical Character Recognition (OCR), documents segment, graphic understanding, and many more.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton.Imagenetclassification with deep convolutional neural networks. In Advances inneural information processing systems, pages 1097–1105, 2012.

[2] Almog, U. YOLO V3 Explained. 2020 October 9.

[3] Anka, A. YOLO v4: Optimal Speed Accuracy for object detection. "A review of a state-of-the-art model for real-time object detection". 2020 May 17.

[4] Augusto Borges Oliveira D, Palhares Viana M. Fast CNN-based document layout analysis. In Proceedings of the IEEE International Conference on Computer Vision Workshops 2017 (pp. 1173-1180).

[5] A. W. Harley, A. Ufkes, and K. G. Derpanis. Evaluation of deep convolutional nets for document image classification and retrieval. In Proceedings of the 2015 13th International Conference on Document Analysis and Recognition (ICDAR), ICDAR '15, pages 991–995, Washington, DC, USA, 2015. IEEE Computer Society.

[6] Barret, Z., et al. "Learning transferable architectures for scalable image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[7] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. arXiv: 2020 April 23.

[8] Brownlee, J. K-Nearest Neighbors for Machine Learning. Machine Learning Mastery, Available from: https://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/, 2018. (Accessed 10 June 2018).

[9] Brownlee, J. Logistic Regression for Machine Learning. Machine Learning Mastery, Available from: https://machinelearningmastery.com/logistic-regression-for-machine-learning/, 2018. (Accessed 10 June 2018).

[10] Brownlee, J. A Gentle Introduction to Transfer Learning for Deep Learning. Deep Learning for Computer Vision. 2017 December 20.

[11] Chowdary GJ, Punn NS, Sonbhadra SK, Agarwal S. Face mask detection using transfer learning of inceptionv3. InInternational Conference on Big Data Analytics 2020 Dec 15 (pp. 81-90). Springer, Cham.

[12] Cs231n.github.io, CS231n Convolutional Neural Networks for Visual Recognition, Available from: http:// cs231n.github.io/convolutional-networks/, 2018. (Accessed 25 September 2018).

[13] Elfouly, S. R-CNN (Object Detection): "A beginners guide to one of the most fundamental concepts in object detection". 2019 July 16.

[14] Eric C Orenstein and Oscar Beijbom. Transfer learning and deep feature-extraction for planktonic image data sets. In 2017 IEEE Winter Conferenceon Applications of Computer Vision (WACV), pages 1082–1088. IEEE,2017.

[15] F. Simistira, M. Bouillon, M. Seuret, M. Wursch, M. Alberti, R. Ingold, ¨ and M. Liwicki, "Icdar2017 competition on layout analysis for challenging medieval manuscripts," in Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 1361– 1370, IEEE, 2017.

[16] Gandhi, R. "R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms".Understanding object detection algorithms. 2018 July 10.

[17] Girshick, R. , Donahue, J., Darrell, T., Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. Tech report (v5): 2014 October 22.

[18] Gong, B., Ergu, D., Cai, Y., Ma, B. A Method for Wheat Head Detection Based on Yolov4. Research square: 2020.

[19] Harrison, O. Machine Learning Basics with the K-Nearest Neighbors Algorithm. Towards data science: 2018 September 11.

[20] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[21] H. Yong, H. Qi and M. Yang, "F-score-like measure: A new measure for spam filtering," International Conference on Machine Learning and Cybernetics, Xian, 2012, pp. 2047-2051, 2012.

[22] Hui, J. Real-time Object Detection with YOLO, YOLOv2 and now YOLOv3. 2018 March 18.

[23] I. Pratikakis, K. Zagoris, G. Barlas, and B. Gatos, "Icdar2017 competition on document image binarization (dibco 2017)," in Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 1395–1403, IEEE, 2017

[24] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, pp. 248– 255, IEEE, 2009.

[25] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440, 2015.

[26] Kane, F. Machine Learning vs. Deep Learning: What's the Difference?. April 2020.

[27] K. Chen, M. Seuret, J. Hennebert, and R. Ingold, "Convolutional neural networks for page segmentation of historical document images," in Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 965–970, IEEE, 2017.

[28] Ke, Q, Boussaid, F. Computer Vision for Assistive Healthcare. 2018.

[29] K. Simonyan, A. Zisserman, Very deep convolutional networks for largescale image recognition, in: International Conference on Learning Representations, 2015, pp. 1–14.

[30] K. Tombre, S. Tabbone, L. Pelissier, B. Lamiroy, and ´ P. Dosch. Text/graphics separation revisited. In Proceedings of the 5th International Workshop on Document Analysis Systems V, DAS '02, pages 200–211, London, UK, UK,2002. Springer-Verlag.

[31] K. Y. Wong, R. G. Casey, and F. M. Wahl. Document analysis system. IBM journal of research and development, 26(6):647–656, 1982.

[32] L. A. Fletcher and R. Kasturi. A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans. Pattern Anal. Mach. Intell., 10(6):910–918, Nov. 1988.

[33] Le QV, Ngiam J, Coates A, Lahiri A, Prochnow B, Ng AY. On optimization methods for deep learning. InICML 2011 Jan 1.

[34] Lei X, Pan H, Huang X. A dilated CNN model for image classification. IEEE Access. 2019 Jul 8;7:124087-95.

[35] Li YD, Hao ZB, Lei H. Survey of convolutional neural network. Journal of Computer Applications. 2016;36(9):2508-15.

[36] Li, M.H., Xu, Y.H., Cui, L., Huang, S.H., Wei, F.R., Li, Z.J., Zhou, M. DocBank: A Benchmark Dataset for Document Layout Analysis. arXiv: 2020.

[37] L. Kang, J. Kumar, P. Ye, Y. Li, and D. Doermann. Convolutional neural networks for document image classification. In 2014 22nd International Conference on Pattern Recognition, pages 3168–3172, Aug 2014.

[38] Long, J, Shelhamer, E, Darrell, T. Fully Convolutional Networks for Semantic Segmentation. CVPR: 2015.

[39] Lu J, Ma C, Li L, Xing X, Zhang Y, Wang Z, Xu J. A vehicle detection method for aerial image based on YOLO. Journal of Computer and Communications. 2018 Nov 7;6(11):98-107.

[40] M. A. Moll and H. S. Baird. Segmentation-based retrieval of document images from diverse collections. In Electronic Imaging 2008, pages 68150L–68150L. International Society for Optics and Photonics, 2008.

[41] M. A. Moll, H. S. Baird, and C. An. Truthing for pixelaccurate segmentation. In 2008 The Eighth IAPR International Workshop on Document Analysis Systems, pages 379– 385, Sept 2008.

[42] M. Diem, F. Kleber, S. Fiel, T. Gruning, and B. Gatos, "cbad: Icdar2017 competition on baseline detection," in 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR), vol. 01, pp. 1355–1360, Nov. 2017

[43] Meel, V. YOLOv3: Real-Time Object Detection Algorithm (What's New?). Deep-Learning: 2021 February 25.

[44] Miracle, R. PANet: Path Aggregation Network In YOLOv4. 2020 July 3.

[45] M. Learning, U. Code, Understanding Support Vector Machine Algorithm From Examples (Along With Code). Analytics Vidhya, Available from: https://www.analyticsvidhya.com/blog/2017/09/understaingsupport-vector-machine-example-code/, 2018. (Accessed 10 June 2018).

[46] Mohan, A. Review On YOLOv3. 2020 June 8.

[47] M. S. Erkilinc, M. Jaber, E. Saber, P. Bauer, and D. Depalov. Page layout analysis and classification for complex scanned documents, 2011.

[48] M. S. Erkilinc, M. Jaber, E. Saber, P. Bauer, and D. Depalov. Text, photo, and line extraction in scanned documents. Journal of Electronic Imaging, 21(3):033006, July 2012.

[49] Nie, B.L. Sun, S.Q. Using Text Mining Techniques to Identify Research Trends: A Case Study of Design Research. Applied Sciences: 15 April 2017.

[50] Oliveira SA, Seguin B, Kaplan F. dhSegment: A generic deep-learning approach for document segmentation. In2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR) 2018 Aug 5 (pp. 7-12). IEEE.

[51] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in International Conference on Medical image computing and computer-assisted intervention, pp. 234– 241, Springer, 2015.

[52] O. Okun, D. Doermann, and M. Pietikainen. Page segmentation and zone classification: The state of the art. LAMPTR-036,CAR-TR-927,CS-TR-4079, 11 1999.

[53] Priyadharshini, N Vijaya, M.S. Genetic Programming for Document Segmentation and Region Classification Using Discipulus. (IJARAI) International Journal of Advanced Research in Artificial Intelligence, Vol. 2, No. 2, 2013.

[54] Redmon J, Farhadi A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767. 2018 Apr 8.

[55] Ren S, He K, Girshick R, Sun J. Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv preprint arXiv:1506.01497. 2015 Jun 4.

[56] Rogers B, Chapman T, Rettele J, Gatica J, Darm T, Beebe M, Dilworth D, Walsh N. Computerized manufacturing of transparent face masks for the treatment of facial scarring. The Journal of burn care rehabilitation. 2003 Mar 1;24(2):91-6.

[57] Rugery, P. Explanation of YOLO V4 a one stage detector. 2020 September 8.

[58] Schott, M. K-Nearest Neighbors (KNN) Algorithm for Machine Learning. Capital One tech: 2019 April 22.

[59] Sharma, P. A Practical Guide to Object Detection using the Popular YOLO Framework – Part III (with Python codes). 2018 December 6.

[60] Supeshala, C. YOLO v4 or PP-YOLO?. 2020 August 23.

[61] Solawetz, J. Breaking Down YOLOv4. 2020 June 4.

[62] S. S. Bukhari, M. I. A. Al Azawi, F. Shafait, and T. M. Breuel. Document image segmentation using discriminative learning over connected components. In Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, DAS '10, pages 183–190, New York, NY, USA,2010. ACM.

[63] Theckedath D, Sedamkar RR. Detecting Affect States Using VGG16, ResNet50 and SE-ResNet50 Networks. SN Computer Science. 2020 Mar;1(2):1-7.

[64] T. M. Breuel, "Robust, simple page segmentation using hybrid convolutional mdlstm networks," in Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 733– 740, IEEE, 2017.

[65] V. P. Le, N. Nayef, M. Visani, J. M. Ogier, and C. D. Tran. Text and non-text segmentation based on connected component features. In 2015 13th International Conference on Document Analysis and Recognition (ICDAR), pages 1096– 1100, Aug 2015.

[66] Y. Xu, W. He, F. Yin, and C.-L. Liu, "Page segmentation for historical handwritten documents using fully convolutional networks," in Document Analysis and Recognition (ICDAR), 2017 14th IAPR International Conference on, vol. 1, pp. 541–546, IEEE, 2017.

[67] Trevino, A. Introduction to K-means Clustering. 2016 December 6.

[68] Tyagi, N. Introduction To YOLOv4. Analytics Steps: 2020 June 25.

[69] Khare, T. Custom Object Detection using Darknet. 2020 June 3.

Appendices

.A Professional Consideration

.A.1 Code of Conduct

As there was not much involvement of privacy data or human experiences, wide ethical implication was not required on this project. However, several ethical issues underlined on BCS Code of Conduct will be taken into consideration wherever they are necessary. The following paragraph consists of the relevance ethical conduct of this project (British Computing Society Code of Conduct, 2015).

.A.2 Section 1 – Public Interest

- Explains that this project was accountable for public health, security, privacy, and the safety of the society and environment. (1a)
- There were not be any form of discrimination on my professional activities. (1c)
- Every individual was gain an equal access towards the benefits of the project. (1d)

.A.3 Section 2 – Professional Competence and Integrity

- All processes and methods through the entire project were done based on my professional ability and competency. (2a)
- This project did not declare anything was out from the professional ability. (2b)
- In order to maintain the quality of the project, it was necessary to have a constant learning mindset to improve the knowledge on the relevant topics of the project. (2c)
- Criticisms and alternatives were fully honored and respected. (2e)
- Prevented any possibility that might cause loss and injury of the society by vindictive, false, or negligent activity. (2f)

.A.4 Section 3 – Duty to Relevant Authority

- All duties will be done with one's professional judgement in accordance with the guidance of the academic supervisor and school requirements. (3a)

- An action will be taken to prevent the rise of conflict of interest with the Relevant Authority. (3b)

.A.5 Section 4 – Duty to the Profession

- Personal duty will keep the good reputation of the profession. (4a)
- Aim to improve the current professional standards. (4b)
- Understand and act based on BCS Code of Conduct, the Chartered Institute for IT. (4c)
- Place integrity and respect as the top priority for professional relationship with all members of BCS and with other members who contribute on the succession of this project. (4d)
- Motivate and encourage fellow members to maintain their professionalism. (4f)

.B Initial Project Plan

Tasks	Activity	Start Date	Predecessor	Time Estimates In Day			%Done	Expected End Date
				Expected	Optimistic	Pessimistic		
1	Project Proposal	12/10/2020	0	3 days	2 days	4 days	100%	16/10/2020
2	Research Related Project	17/10/2020	0	5 days	4 days	7 days	100%	23/10/2020
3	Interim Report	28/10/2020	2	10 days	8 days	12 days	100%	12/11/2020
4	Gathering Requirements	14/11/2020	3	7 days	5 days	10 days	30%	25/11/2020
5	Coding	26/11/2020	4	50 days	40 days	60 days	0%	27/01/2021
6	Testing the Code	28/01/2021	5	6 days	5 days	7 days	0%	04/02/2021
7	Analysing the Data	05/02/2021	6	6 days	5 days	7 days	0%	12/02/2021
8	Improving the Code	13/02/2021	7	12 days	10 days	15 days	0%	28/02/2021
9	Project Poster	01/03/2021	8	8 days	6 days	10 days	0%	11/03/2021
10	More Improvement on Code	12/03/2021	7	10 days	8 days	12 days	0%	25/03/2021
11	Draft Report	26/03/2021	10	22 days	20 days	25 days	0%	20/04/2021
12	Final Report	21/04/2021	11	18 days	16 days	20 days	0%	11/05/2021

FIGURE 1: Initial project plan

.C Meeting Log

- Meeting 1 (22/09/2020): Proposing the project idea and the scope of exploration to supervisor and discuss about the proposal and interim report deadlines.
- Meeting 2 (26/10/2020): General discussion about literature review for interim report, and some strategies approach from background research that might be useful.
- Meeting 3 (6/11/2020): Discussing about the progression of the interim report and receive feedback on how and where to get the dataset from.
- Meeting 4 (1/02/2021): A quick follow-up about the current progress
- Meeting 5 (22/02/2021): Discussing about the poster and asking for feedback.
- Meeting 6 (26/03/2021): Showing results on Yolo-V3 and asking for feedback to increase the performance of the current model.
- Meeting 7 (19/04/2021): Showing the classes of the labeling data, and a quick follow-up of the Yolo-V3.
- Meeting 8 (26/04/2021): Yolo-V4 showing better results than Yolo-V3.
- Meeting 9 (03/05/2021): Discussing about creating a user interface for converting pdf file to images format.
- Meeting 9 (10/05/2021): Feedback on current draft report.
- Meeting 9 (17/05/2021): Asking several questions about the final report.