

Application of Machine Learning to Automate Classification and Information Extraction in Industrial
Construction Documents

by

Narges Sajadfar

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Construction Engineering and Management

Department of Civil and Environmental Engineering

University of Alberta

© Narges Sajadfar, 2022

Abstract

Industrial construction projects are usually mega-projects that involve millions of labour person-hours and generate hundreds of thousands of documents. Construction documents represent a vital source of information and knowledge regarding the project scope. Documents come in different types and include structured information such as data tables and unstructured information such as text, images, and drawings. The documents may consist of contract forms, drawings that define the quantities and qualities of materials, standards, and specifications required to carry out the project. Documents usually involve multi-versions and address different systems in a project, such as architectural, structural, electrical, and mechanical systems. The ability to extract and organize structured and unstructured information from these documents is a time-consuming process that is critical for effective project control and decision making. This task is more challenging and labour-intensive when documents are provided in image formats requiring human intervention to extract the required information. The objective of this research is to address this challenge by introducing an automated approach for managing and extracting information from construction documents. This research describes the development of automatic classification and information extraction based on both the text and images in industrial construction documents. The development of the proposed method includes the testing of various deep learning classification algorithms, to identify suitable models for construction documents.

The results of the research confirmed the effectiveness of machine learning algorithms for classifying and extracting information from unstructured construction documents with limited text. This dissertation makes a major contribution by presenting a high-precision classification approach for construction documents that incorporates scanned images, with different sizes and resolutions. Furthermore, the method of automatic title block detection was demonstrated for unstructured construction documents in this research.

Preface

This thesis is an original work by Narges Sajadfar.

A version of Sections 5.1.3 and 5.1.4 is published as “Text detection and classification of construction documents.” ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction. Vol. 36. IAARC Publications, 2019. Pages 446-452.

DOI: <https://doi.org/10.22260/ISARC2019/0060>

Acknowledgments

I want to express my sincerest appreciation to my supervisor, Dr. Yasser Mohamed, whose efforts, and support made this dissertation possible. He provided me with endless encouragement and expert guidance. It was my honour and pleasure to be his student.

Next, I want to thank my valued dissertation committee members, Dr. Ahmed Hammad and Dr. Nilanjan Ray, for providing me with constructive comments and suggestions, which significantly improved the quality of my research.

Also, I express my deepest thanks to PCL Industrial Management Inc. and Rick Hermann for providing me the opportunity to do my research in the construction industry and gain valuable experiences.

My special thanks go to Abhineet Singh for his knowledgeable advice and guidance. He was very generous in sharing his experiences on object detection methods and algorithms.

I would also like to take this opportunity to thank my colleagues, Sina Abdollahnejad for collaboration on the document classification model using TF-IDF, Parinaz Jafari and Osama Mohsen for their guidance and support.

I am also thankful to my family for all their love and unconditional support throughout my life.

Table of Contents

Abstract.....	ii
Acronyms/Abbreviations	x
Chapter 1 Introduction	1
1.1 Research background	1
1.2 Problem identification.....	4
1.3 Research objectives.....	5
1.4 Expected contribution	5
1.5 Thesis organization	6
Chapter 2 Literature review	7
2.1 Construction document classification	7
2.2 Image processing and object detection	9
2.3 OCR technique.....	12
2.4 Machine learning for document classification.....	13
2.5 Machine learning and deep learning algorithms	19
Chapter 3 Research methodology	21
Chapter 4 Data collection and analysis.....	23
4.1 Data set preparation	23
4.2 Pre-processing steps for improving image quality.....	32
4.3 Layout analysis	33
Chapter 5 Design and implementation.....	34
5.1 Phase 1: Developing classification based on text	34
5.1.1 OCR technique and text extraction	35
5.1.2 Classification based on TF-IDF and different algorithms.....	39
5.1.3 Classification based on a pre-defined set of keywords	54
5.1.4 Classification based on deep learning–LSTM	56
5.1.5 Comparison of results	58
5.2 Phase 2: Developing classification based on image.....	59
5.2.1 Classification based on TensorFlow object detection API.....	60
5.2.2 Classification based on AlexNet	72
5.2.3 Comparison of results	79
5.3 Phase 3: Developing title block detection and information extraction	81
5.3.1 Proposed Methodology of title block detection	83
5.3.2 Experiments	85
5.3.3 Implementation and training.....	89

5.3.4 Evaluation and results	91
Chapter 6 Test of Title Block Detection and Information Extraction Model	95
6.1 Test of title block detection on drawings documents.....	95
6.2 Test of title block detection on non-drawings documents	108
6.3 Test of information extraction model.....	115
Chapter 7 Selected solution	118
7.1 Comparing the models	120
7.2 Evaluating the selected solution.....	123
7.3 Classification results	124
Chapter 8 Conclusion.....	127
8.1 Conclusion	127
8.2 Contribution	128
8.3 Limitations and recommendations for future work.....	129
References.....	131
Appendix A- Title block detection.....	141

List of Tables

Table 4-1. Percentage of document types	25
Table 4-2. Percentage of drawing and non-drawing documents.....	26
Table 5-1. TF-IDF samples for one document.....	47
Table 5-2. Accuracy of the four classification algorithms	52
Table 5-3. Precision results of the four classification algorithms.....	52
Table 5-4. Recall results of the four classification algorithms.....	53
Table 5-5. keywords of ten classes	55
Table 5-6. LSTM architecture.....	57
Table 5-7. Data set of Test 1	62
Table 5-8. The architecture of Test 1	63
Table 5-9. Data set of Test 2	65
Table 5-10. The architecture of Test 2	65
Table 5-11. Data set of Test 3	69
Table 5-12. The architecture of Test 3	69
Table 5-13. Data set of Test 1	73
Table 5-14. The architecture of Test 1	73
Table 5-15. Data set of Test 2	75
Table 5-16. Result of Test 2.....	75
Table 5-17. Data set of Test 3	77
Table 5-18. Result of Test 3.....	77
Table 5-19. Summary of the result of TensorFlow API and Alex Net	80
Table 5-20. Title block detection results.....	93
Table 6-1. The architecture of Test 1	96
Table 6-2. The architecture of Test 2.....	96
Table 6-3. The architecture of Test 3.....	97
Table 6-4. The architecture of Test 4.....	101
Table 6-5. The architecture of Test 5.....	105
Table 6-6. Summary of Results for title block detection on drawing documents.....	107
Table 6-7. The architecture of Test 1	108
Table 6-8. The architecture of Test 2.....	109
Table 6-9. The architecture of Test 3.....	110
Table 6-10. The architecture of Test 4.....	111
Table 6-11. The architecture of Test 5.....	112
Table 6-12. Summary of result for title block detection on non-drawing documents.....	114
Table 6-13. Result of information extraction.....	116
Table 7-1. Classification result test 1.....	120
Table 7-2. Classification result test 2.....	121
Table 7-3. Classification result test 3.....	121
Table 7-4. Confusion matrix	125
Table 7-5. Classification result	126

List of Figures

Figure 1-1. Samples of construction documents: 1) Data sheet, 2) Isometric drawing, 3) Bill of materials	1
Figure 1-2. Workflow of document and data management.....	2
Figure 1-3. Industrial document log.....	3
Figure 2-1. SVM for two-class classification	16
Figure 3-1. Overall research methodology	22
Figure 4-1. Document types in the first dataset	24
Figure 4-2. Sample of drawing documents	28
Figure 4-3. Sample of non-drawing documents.....	30
Figure 4-4. Bill of materials layouts	31
Figure 4-5. Steps of image enhancement	32
Figure 4-6. Possible locations of the table of information	33
Figure 5-1. Proposed method of construction document classification based on text	34
Figure 5-2. Steps of Connected Component analysis	36
Figure 5-3. Sample code of connected component in Matlab.....	37
Figure 5-4. The process of text extraction	38
Figure 5-5. Example of text extraction steps	39
Figure 5-6. Methodology of classification based on TF-IDF and different algorithms.....	41
Figure 5-7. Dataset samples for classification based on TF-IDF and different algorithms	42
Figure 5-8. Example of cropping the images.....	44
Figure 5-9. Sample results of cropping the image, text extraction, text cleaning, and tokenization.	46
Figure 5-10. K-fold cross-validation.....	48
Figure 5-11. Sample of the training dataset in CSV file.....	49
Figure 5-12. Dataset samples for Classification based on a pre-defined set of keywords.....	54
Figure 5-13. LSTM training progress	58
Figure 5-14. Methodology of construction document classification based on image.....	59
Figure 5-15. Sample of labelling.....	62
Figure 5-16. The result of the model on isometric drawing.....	64
Figure 5-17. The result of the model on the layout drawing.....	64
Figure 5-18. The result of the model on the datasheet.....	66
Figure 5-19. The result of the model on isometric drawing.....	67
Figure 5-20. The result of the model on the work package	68
Figure 5-21. The result of the model on the schematic.....	70
Figure 5-22. The result of the model on Bill of Materials	71
Figure 5-23. The result of the model on the cable schedule	71
Figure 5-24. Training progress of Test 1	74
Figure 5-25. Result of Test 1	74
Figure 5-26. Training progress of Test 2	76
Figure 5-27. Result of Test 2	76
Figure 5-28. Training progress of Test 3	78
Figure 5-29. Result of Test 3-A).....	78
Figure 5-30. Result of Test 3-B).....	79
Figure 5-31. Different stages of developing title block detection and information extraction	81
Figure 5-32. The proposed methodology of title block detection and information extraction	83
Figure 5-33. Construction document example and location of the title block	84

Figure 5-34. Result of the three image resizing options	87
Figure 5-35. Sample of Manual labelling 1	88
Figure 5-36. Sample of Manual labelling 2	89
Figure 5-37. Data preparation and training steps	90
Figure 5-38. Example of result on Tensorboard	90
Figure 5-39. Sample of title block detection	91
Figure 5-40. Sample results of failed title block detection	92
Figure 5-41. Sample results of title block detection and information extraction	94
Figure 6-1. Result of Test 3	97
Figure 6-2. Sample of result that has two detected bounding boxes	98
Figure 6-3. Sample of the noisy image	98
Figure 6-4. Layout Drawing	99
Figure 6-5. Schematic Drawing	99
Figure 6-6. Sample of divided images	100
Figure 6-7. Result of Test 4 – Isometric Diagram	101
Figure 6-8. Result of Test 4 - Layout	102
Figure 6-9. Result of Test 4 - Schematic	103
Figure 6-10. Result of Test 4 - Wiring Diagram	104
Figure 6-11. Result of Test 5- Isometric drawing- A)	105
Figure 6-12. Result of Test 5- Isometric drawing- B)	106
Figure 6-13. Result of Test 1- Datasheet	109
Figure 6-14. Result of Test 2- work package	110
Figure 6-15. Result of Test 3- Bill of Materials	111
Figure 6-16. Result of Test 4- cable schedule	112
Figure 6-17. Result of Test 5- work package	113
Figure 6-18. Result of Test 5- datasheet	113
Figure 6-19. Overview of the information extraction model	115
Figure 6-20. Table of information- A)	117
Figure 6-21. Table of information- B)	117
Figure 7-1. Sample of inputs	118
Figure 7-2. Process of three tests	119
Figure 7-3. Recall results	122
Figure 7-4. Precision results	122
Figure 7-5. Evaluation method	123
Figure 7-6. Document types in the second dataset	124

Acronyms/Abbreviations

AEC	Architecture, Engineering & Construction
API	Application Programming Interface
CAD	Computer-Aided Design
CBIR	Content-Based Image Retrieval
CICS	Construction information classification systems
CNN	Convolutional Neural Network
DPI	Dots Per Inch
ED	Elevation Datum
IE	Information extraction
LSTM	Long Short-Term Memory
LSVC	Linear Support Vector Classifier
ML	Machine Learning
MSER	maximally stable extremal regions
NLP	Natural Language Processing
NN	Neural Network
OCR	Optical Character Recognition
PDF	Portable Document Format
R-CNN	Region Based Convolutional Neural Networks
RNN	Recurrent Neural Network
ROI	Region of Interest
SSD	Single Shot Detector
SVM	Support Vector Machine
TF-IDF	Term Frequency-Inverse Document Frequency
TFRecord	TensorFlow Record
XML	Extensible Markup Language
YOLO	You Only Look Once

Chapter 1 Introduction

1.1 Research background

Everyday construction companies receive hundreds of documents. Many documents including images and drawings are generated along with the different phases of construction projects, and they represent a rich source of information and knowledge. Construction companies procure thousands of documents that include structured and unstructured data and vary in terms of purpose, type, content, and format [1]. Figure 1-1 shows samples of construction documents. The first document includes a table and text, and the table of information is located at the bottom. The second document is a drawing containing images and text, and the table of information is found on the right side. The third document includes a table and text, and the table of information is located at the top of the page. The samples shown in Figure 1-1 demonstrate different structures and formats of construction documents. Having a large number of documents can create a potential problem in industrial projects as companies have to spend more time on information management and retrieval, which can increase processing time and data access, increase misused data and errors, and decrease the performance and collaboration of team members. Managing and grouping construction documents, especially images and drawings, is challenging for construction companies.

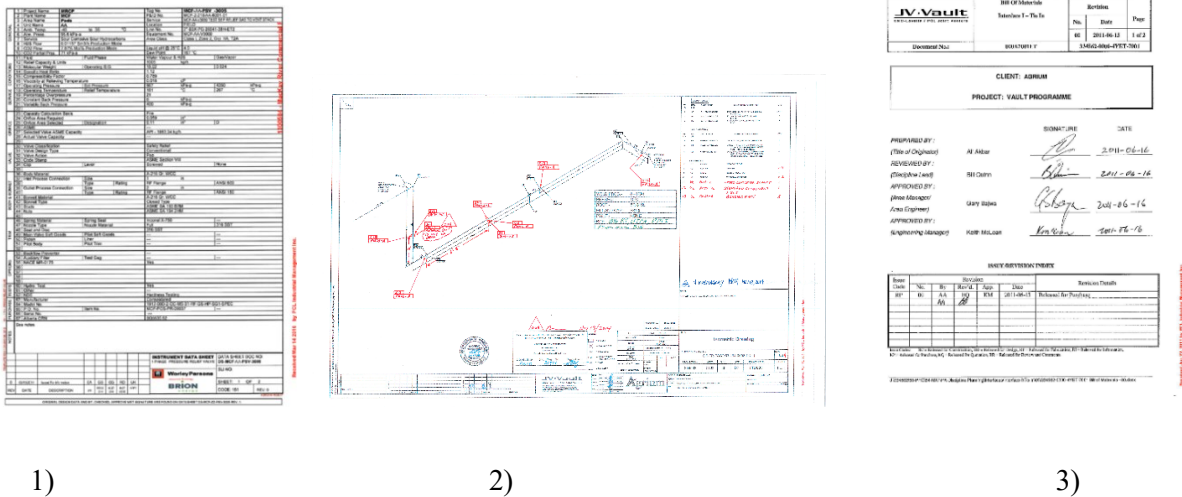


Figure 1-1. Samples of construction documents: 1) Data sheet, 2) Isometric drawing, 3) Bill of materials

Based on ISO 9001:2015, document control requires the following activities: distribution and retrieval, storage and preservation, control of document changes, retention and disposition [2]. Document control begins with the reception of a document from an internal or external department. After receiving the document, it will be reviewed manually. Manual document review includes verification details of documents and document evaluation. The approved document will be sent to the archive. The store part of the process contains data collection and classification, storing and managing data collection, protection, and data maintenance. The retrieval part of the process will provide access to data based on access restrictions. The distribution part of the process will distribute the information based on the distribution list. The disposition part of the process will determine how long the data must be kept, and finally, data will be destroyed according to business, legal, and regulatory requirements. Figure 1-2 shows the workflow of document control based on ISO 9001:2015 documents control conditions.

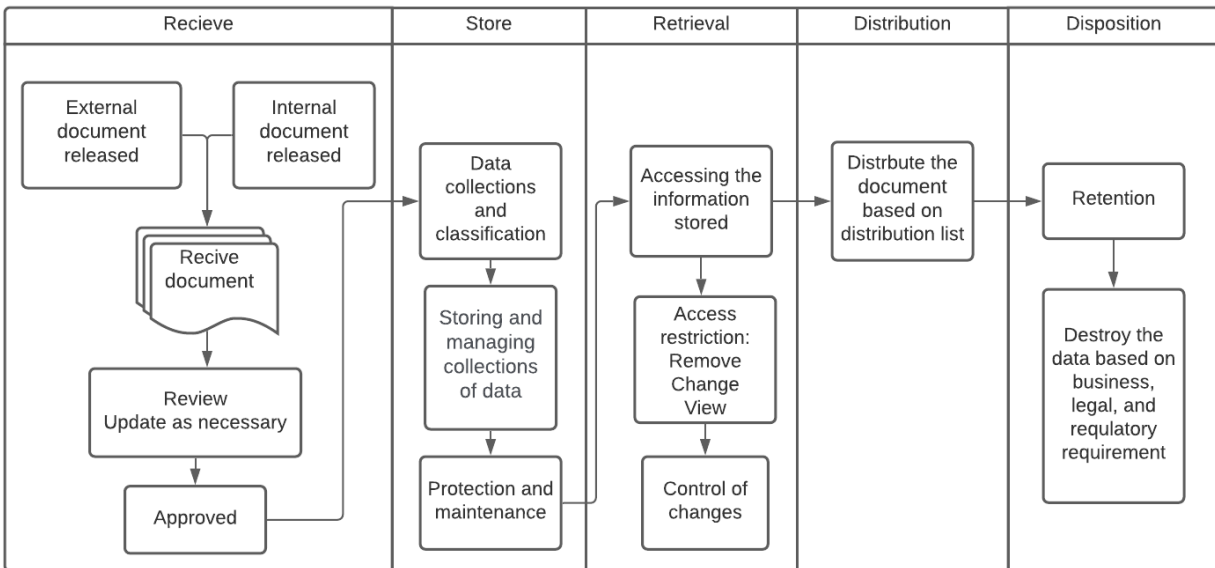


Figure 1-2. Workflow of document and data management

However, each company develops its document and data management process based on the organization's size, types of activities, processes, products, and services. For example, the document controller of a construction company will receive the transmittal letter and batch of construction documents each time.

Transmittal document or letter includes the details of the document information sent, such as document name, revision, and description. Document controllers need to check both transmittal letters and construction documents to ensure they are matched and correct. Sometimes information is missing in the transmittal letter. The document controller needs to open each document, visually recognize what appears in such documents, and label/describe them. Then, according to document label(s), they will get categorized under different classes. The document controller will update the document management system based on the received information in the last steps. Figure 1-3 shows a sample of an industrial log system, which includes document type, number, ID, revision, title, location, date received and created.

Drag a column header here to group by that column											
Latest ...	Document Type	Document Number	Revi...	Title	Location	Date Received	CONTRACT...	Date Created	Document ID	File Original	TRANSMITTAL RECEIVED
	STANDARD DETAILS	074-0000-070-000-005 02	0	Differ...	\PCLT...	9/24/2009	Contract	9/24/2009	07400000700...	\PCLINC....	IFCC-10218: K2SM-PC-T-00946
	DETAILS	074-1500-070-000-008 07	0	1500-...	\PCLT...	6/23/2010		6/23/2010	07415000700...		
	DATA SHEET	KE40057P-F06-0006-002	001	INSTR...	\PCLT...	12/4/2012	Post Contract	12/6/2012	KE40057PF06...	\PCLINC....	K2SM-PC-T-00212
	TERMINATION DE...	KE40013P-E12-0006-001	001	INSTR...	\PCLT...	7/26/2012	Post Contract	9/24/2012	KE40013PE12...	\PCLINC....	K2CM-PC-T-00189
	PROJECT INSTRU...	PI-P-180-0474	0	INSTA...	\PCLT...	10/30/2014	Post Contract	10/30/2014	PIP1800474	\PCLINC....	
	DETAILS	KE40057P-E01-0007-004	001	LAKES...	\PCLT...	7/20/2012	Post Contract	7/20/2012	KE40057PE01...	\PCLINC....	K2CM-PC-T-00205
	DATA SHEET	1610-833-ZIT-7450-3	-	INSTR...	\PCLT...	5/7/2014	Post Contract	5/12/2014	1610833ZIT7...	\PCLINC....	K2SC-PC-T-00298
	DATA SHEET	KE40099P-F06-0025-001	001	2600-...	\PCLT...	7/30/2012	Post Contract	10/9/2012	KE40099PF06...	\PCLINC....	K2CM-PC-T-00194
	LOOP DIAGRAM	KE40061P-E02-0001-006	001	4900...	\PCLT...	3/28/2013	Post Contract	4/2/2013	KE40061PE02...	\PCLINC....	K2SM-PC-T-00401
	DETAILS	074-1600-070-000-043 04	0	1600-...	\PCLT...	6/16/2010	Contract	6/16/2010	07416000700...	\PCLINC....	
	TERMINATION DE...	KE40057P-E12-0002-061	001	INSTR...	\PCLT...	7/10/2012	Post Contract	7/19/2012	KE40057PE12...	\PCLINC....	K2CM-PC-T-00184
	TERMINATION DE...	KE40057P-E12-0002-026	001	INSTR...	\PCLT...	7/11/2012	Post Contract	7/16/2012	KE40057PE12...	\PCLINC....	K2CM-PC-T-00187

Figure 1-3. Industrial document log

While manual or database project document control is straightforward, significant effort is usually spent on document classification. Several new approaches are being investigated to solve this problem, such as the automated information retrieval approach [3] and automated document classification. Those studies show that a potential solution is to develop an automatic classification of construction documents. Finding a structure for automatic classification of construction documents, especially images and drawings, can increase the efficiency of a company's information management process and increase the company's performance and productivity. Automatic document classification and information extraction can bring lots of benefits to the industry. Also, it can improve the quality of the document control process by increasing

accuracy and efficiency, avoiding error, and faster document control. Automatic document classification is a solution for better data management which can create competitive advantages for companies [4].

1.2 Problem identification

Although automatic classification has excellent advantages for construction companies, the successful implementation of this method is not an easy task. The main problem is that the data set is unstructured and not in the standard format. Different methods and algorithms need to test to find the best solution.

There are three main ways to do classification: text-based, image-based, and a combination of both. Each technique has its problems. For example, text-based classification suffers from issues like the quality of document prints and layout and formatting used in the document. The results are questionable when the text-based approach is applied in the context of drawing documents that have limited text.

In image-based methods, similar problems like quality and resolution of images or mixing of texts with images can cause difficulties in terms of accurate classification. The result of an image-based approach on a document that has only text such as a report or bill of materials is unknown. From a more detailed point of view, the problems can be summarized as belonging to the following categories:

- Different document layouts, formats, sizes, and quality
- Mixed text and image documents
- Some documents generated as scans of hard copies with handwritten annotations and noises in the file
- Some documents generated as PDF with only images from an authoring CAD application with limited text
- Low quality and resolution
- Difficulty in extracting high accuracy text
- The high degree of similarities between some document types

1.3 Research objectives

The main goal of this study is to investigate how machine learning, deep learning techniques and algorithms can be applied in construction document classification in an efficient and accurate way. The study will focus on three main objectives:

1. Design and implementation of different methods for automatic classification of documents related to industrial construction projects. Under this objective, alternative approaches and algorithms will be evaluated to enable the classification of construction documents based on their content, including a mix of text, tables, drawings, and images.
2. Investigate suitable methods for automated extraction of information from construction project documents. In particular, the research will focus on industrial project documents generated during construction phases and will aim to evaluate different methods for the automated classification and extraction of information.
3. Evaluate the selected solution on a larger scale of datasets to assess its practicality and determine whether it works for the construction domain.

1.4 Expected contribution

From an academic point of view, this research is one of the few studies that specifically use machine learning and deep learning algorithms for the classification of industrial construction documents. As a result of the proposed framework, comprehensive automated document classification is expected which can perform accurate classification on scanned construction documents, with different templates, sizes and resolutions, and limited text. The comprehensive document classification includes the classification of all types of documents such as drawing and non-drawing documents. Another expected contribution of this study is automated title block detection and information extraction. A novel method for title block detection on unstructured construction documents was proposed by using object detection API.

From an industry point of view, automating the process of classification construction documents is a useful tool for effective construction documents control. The outcome of this study will be helpful for improving the document control practice at the collaborating company. Also, the successful completion of this research can contribute to increased productivity and decreased cost of construction document management practices.

1.5 Thesis organization

Chapter 1 briefly introduces the project's background, problem statement, research objectives, and research organization. Chapter 2 reviews the technologies employed in this project, including text and image classification, image processing, OCR technique, machine learning, and deep learning algorithms. Chapter 3 describes the overall research methodology. Chapter 4 describes the data collection and analysis tasks. Chapter 5 describes the design and implementation task. Chapter 6 describes different tests designed for the title block detection model. Chapter 7 compares different solutions and evaluates the selected solution on a large-scale dataset. Chapter 8 includes the research contributions, limitations, and recommendations for future work.

Chapter 2 Literature review

Undoubtedly, document control affects all aspects of a construction project, and many companies struggle to maintain effective document control. Document control is receiving all the documents, and after approval, it needs to classify and storage them based on their document type. Construction document classification is part of document control which has an essential role in the efficiency of document control. The literature review consists of five topics. The first presents the review of construction document classification, which uses text-based and image-based classification. The second topic is image processing. The third topic is about the OCR technique, which is the central part of the text-based classification. The fourth topic describes the machine learning algorithms which was used in this research for document classification. The fifth topic discusses the use of machine learning and deep learning algorithms in the construction domain.

2.1 Construction document classification

The automatic classification of documents into pre-defined categories is a well-established topic in the machine learning domain. The literature review shows that several automatic classifications of construction documents were developed. Kang and Paulson [5] analyzed practical civil engineering projects through the life cycle of a project. They provided a framework of information classification based on a construction information classification system (CICS). Caldas et al. [6] introduced machine learning methods to classify construction documents. They applied different feature selection and classification algorithms on construction project documents. According to their report, the combination of normalized term frequency-inverse document frequency (TF-IDF) weighting as a feature selection method and support vector machine (SVM) as a machine learning algorithm provided 91% accuracy. Caldas et al. [7] introduced a framework of hierarchical document classification for construction management information systems in three levels, and a variety of machine learning algorithms were applied. Their result shows that hierarchical classification is more complicated than flat classification and the accuracy also is lower than flat

classification. Mahfouz [8] used support vector machines (SVM) algorithm to classify unstructured construction documents that included correspondences, meeting minutes, and claims. The proposed model used 475 documents and had accuracy ranging between 91% and 83%. Hsu [9] used content-based text mining techniques to extract the text of computer-aided design (CAD) documents and create an indexing database. Then, they applied the vector space model (VSM) algorithm for similarity matching and retrieval of documents. Salama and El-Gohar [10] developed an automated compliance checking (ACC) in construction. They used a machine learning-based text classification algorithm for clauses and sub-clauses of general conditions, specifically construction contracts. Also, they used a feature selection algorithm for training and testing the sub-clauses. They proposed semantic text classification by combining TF-IDF and SVM algorithms to classify clauses and sub-clauses of general conditions in construction contract documents. Overmann et al. [11] Used a new method for feature selection: the term frequency-inverse class frequency-class frequency (TF-ICF-CF) and vector space model (VSM) based algorithm for classification of content components in technical communication. Their model achieved 84% accuracy on average, 90% for high-quality document classification.

Although previous studies have demonstrated the effectiveness of text classification techniques in the case of construction documents, these studies focused mainly on text-rich documents such as contracts, correspondences, emails, etc. In addition, most of the studies rely on well-formed documents where text is readily available in machine-readable format (e.g., PDF, doc, or CAD files). On the other hand, construction drawings are not text-rich and only contain fragmented chunks of text and numeric data that are used mainly to annotate a drawing or populate information in the drawing title block. In addition, in many cases, such documents are ill-formed due to the fact they are generated as scans of hard copies with handwritten annotations and noises in the file, or they are generated as PDF with only images from an authoring CAD application to protect the content for legal or IP purposes. Such conditions make the text content of such documents very poor and fragmented.

Many researchers have investigated text classification. However, a few of them focused on construction document classification when they involved text in addition to drawings and images. Caldas et al. [6] introduced a prototype for a construction document classification system. They defined standard classification structures, called construction information classification systems (CICCSs), which create concept hierarchies that can be used for text classification. Later, they developed a prototype program for automatic hierarchical classification of construction project documents based on project components [7]. Al Qadi et al. [12] used natural language processing (NLP) tools for text analysis and proposed a hybrid approach for automatic clustering based on text similarity. This study developed a core cluster and trained a text classifier on core clusters for classifying other documents [13]. In addition, there is quite a bit of study about using image processing techniques for construction performance monitoring. Golparvar-Fard and Peña-Mora [14] introduced vision-based methods for construction process monitoring, and extracted building information from images to analyze the progress status. Golparvar-Fard et al. [15] extracted as-built semantic information from 3D CAD software. Wang and Cho [16] introduced intelligent scanning and visualization of dynamic construction and used image-based object recognition and tracking algorithms to achieve their goal.

2.2 Image processing and object detection

Image processing has been an active research domain since the 1970s [17]. During the 1990s, digital technology was introduced, and the number of potential uses of digital images has increased enormously since then. Many researchers worked on retrieving images and introduced a variety of techniques and tools. For example, Rorvig et al. [18] presented a pattern for image classification by feature matching. Flickner et al. [19] announced Query by Image Content system to explore content-based image retrieval. Bach et al. [20] provided an image search engine based on visual features such as colour and shape. Traditionally, there are three basic ways of image processing: content-based image retrieval (CBIR), image retrieval by text, and hybrid image retrieval. Content-based image retrieval (CBIR) creates the raw information from features

such as lines, edges, angles, colour, and patterns to extrapolate a meaning for images [21]. Finally, hybrid image retrieval uses image and text retrieval to increase the system's capability [22]. Each of these three ways can be used for image clustering. However, the majority of researchers are using (CBIR) for image clustering. There are lots of studies about image clustering methods and techniques. For instance, Chen et al. [23] introduced a novel content-based image retrieval scheme by unsupervised learning approach. They used the dynamic clustering method on the image retrieval scheme. Le Saux and Boujemaa [24] used a fuzzy logic algorithm for content image clustering, and their algorithm was relied on an unsupervised databased category. In recent years, most researchers have been using fuzzy logic, machine learning, and deep learning for image processing [25–27].

Object detection has been an active research domain in computer vision for several decades. It refers to a collection of related tasks for detecting and classifying certain objects in digital images and videos [28]. Image classification and object localization are two important computer vision tasks that constitute this process. Image classification involves classifying an image based on its semantic contents, while object localization is the process of finding all objects of interest in an image and drawing bounding boxes around their locations [29]. The accuracy of object detection has increased significantly since the advent of deep learning [30,31]. There are several deep learning-based approaches to object detection, which can be divided into region proposal-based and regression-based, also known as two-stage and single-stage detectors in the literature. Region proposal-based detectors include R-CNN [32], Fast R-CNN [33], Faster R-CNN [34], and Mask R-CNN [35] while SSD [36], YOLO [37] and RetinaNet [38] are popular examples of regression-based detectors. There is also a significant body of work on applying object detection methods for text detection and recognition. For example, these detectors have been widely used for text region detection and text recognition purposes [39–42]. Liu et al. [43] introduced Markov Clustering Network (MCN), an object detection method based on graph clustering that can detect text objects in various text sizes and orientations. Liao et al. [44] proposed rotation-sensitive regression detector (RRD) that performs oriented object detection to increase the accuracy of text detection. Nagaoka et al. [45] introduced an end-

to-end text detection method based on Faster R-CNN that generates regions of interest (ROIs) through multiple region proposal networks (RPNs) and uses feature maps from multiple convolutional layers.

In addition, object detection techniques have greatly improved the performance of table detection and recognition. Hao et al. [46] defined horizontal and vertical ruled table area and used CNN for table detection in PDF documents. Gilani et al. [47] used Faster R-CNN as an image transformation to separate text regions and white spaces present in the document image followed by Faster R-CNN for table detection. Arif and Shafait [48] also used Faster R-CNN for table detection by using foreground and background features. The current work likewise uses Faster R-CNN [34] for table detection as well as localization of the title block. The text inside this table is then used for classification and information extraction.

Previous work that is discussed above focused on scene text [43–45], e-book document [46], and research paper, magazine, and news [47,48]. In this research, I applied similar techniques to the new domain of construction documents. Construction documents include technical drawings such as plans and design details and non-drawing documents such as reports, bill of materials, specifications, etc. In construction literature, there is minimal research about title block detection. Najman et al. [49] was one of the first comprehensive works about title block detection. They introduced automated title block detection in technical drawings based on signal measurements. Their method was able to find the drawing format matching the size and template. Also, Najman et al. [50] improved their previous model by adding a rectangle finding algorithm to detect the title block. Their method achieved about 70% recognition accuracy for title blocks in technical drawings. Cao et al. [51] applied layout analysis to detect rectangular regions and compared them to pre-defined patterns to find the title block in engineering drawings. Ondrejcek et al. [52] used manual title block detection and template mapping for information extraction from scanned engineering drawings, which is time-consuming. Banerjee et al. [53] introduced the automatic creation of hyperlinks in construction documents, and used the lower right corner of a document as the fixed location for the title block to extract the sheet number. Their model was only applied to drawing documents. Previous work in the construction domain related to title block detection has focused mainly on drawing documents

and did not achieve practical accuracy. This research presents a title block detection approach for all types of construction documents including technical drawing and non-drawing and aims for a higher accuracy that enables effective automation of this task. Although much research has been undertaken in image processing, there is still a gap in the context of construction document classification, especially when documents include text and drawings/images. Outside the construction domain, image processing and object detection are thoroughly researched topics with many advancements that can be adapted to the needs of the construction industry.

2.3 OCR technique

Optical character recognition (OCR) is a process that can convert the scanned text and image into a machine-readable document. Also, it is one of the most used extraction techniques for documents and images [54]. The literature review shows that previous researchers used OCR in construction documents for different purposes. Berkahn and Tilleke [55] used OCR and Koheren neural networks in construction drawings to extract information about the dimensions of construction parts and inscription texts. Their model checks all the dimension line points and construction element points to extract the dimension number and text. However, the user needs to check the result and correct the errors. Banerjee et al. [56] used the OCR system to hyperlink engineering drawing documents. They created a hyperlink based on the extracted information. As a result, the engineers can quickly navigate between different files. They achieved more than 94% accuracy on automatic hyperlinking. Also, Banerjee et al. [57] used the OCR engine in the architecture, engineering and construction (AEC) industry drawing documents for detection of elevation datum (ED) name and graphical shape of ED; also they used experimental analysis to validate the ED name. The result of their research shows they achieved an overall accuracy of 95% for ED detection and accurate destination document text recognition. Banerjee et al. [58] used the OCR engine for extraction of alphabetic code and text of reference document to create automatic navigation among architectural and construction documents. Their result shows that OCR has more than 91% accuracy on character level recognition. Seraogi et al. [59] used the OCR engine in AEC to find the correct orientation of the documents based on

the information of extracted texts and graphical shapes. They used mixed text/image drawings as their case study and achieved more than 99% accuracy on automatic orientation. Gupta et al. [60] used the OCR engine in AEC to extract the title of the documents. In their method, the OCR engine scans the information table only to extract the title. Also, they used historical data to increase the accuracy of their model. However, the user should review the extracted title to achieve 100% accuracy.

Also, several commercial software packages are using OCR technology to extract document information. Procure is a construction project management software that uses OCR on a pre-defined template to extract drawing numbers, drawing disciplines, and drawing titles. Drawing block text should be on the bottom right of the drawing with a specific size and location; then, the Procure can automatically pre-populate the fields [61]. Docparser is another software which is using an OCR engine to extract the text from any document. The user must define the specific locations inside the document and rules to apply to all documents. Then Docparser will train to find the place of each field. Finally, this software will extract the text from a pre-defined location based on regular expressions and pattern recognition [62]. Microsoft Azure is a computer application using various technologies such as an OCR engine for text analysis. It can extract the text from images and documents, which can be used for label recognition, key phrase extraction, and enable searching. Microsoft Azure is also using computer vision algorithms for image classification. The user must provide the labelled images to train a custom vision algorithm and create a model to classify new images [63]. The biggest problem of existing software is that they are not suitable for documents with inconsistent template formats, and their accuracy will significantly decrease in such situations.

2.4 Machine learning for document classification

An automatic document classification system uses machine learning algorithms to classify documents according to predetermined categories [64]. Then, each document will be classified into single or multiple classes. Based on the fact that each document's class was known, supervised learning was applied in this research. By training on a set of documents, supervised learning can then predict the categories of

documents. The classification of documents can then be effectively used as a management and sorting tool. Machine learning algorithms are widely used for document classification, and several of them have been applied in the present study. I will discuss each of them in detail.

Naïve Bayes

Naïve Bayes is a method for supervised learning that is based on Bayes' rule and a probabilistic classifier. The assumption is that the data are independent. Many problems can be solved with Naïve Bayes, including multi-class prediction and sentiment analysis. It has higher accuracy compared to other algorithms like Decision Tree and k-NN even though it is trained on a smaller amount of data. Naïve Bayes has several advantages: This algorithm is fast, scalable, suitable for continuous and discrete data. Also, it does not require a large amount of training data set. It can be used for binary classification and multi-class classification.

Two modifications to naïve Bayes can be used for text classification: the multi-variate Bernoulli and the multinomial model [65]. The multi-variate Bernoulli assumes each feature has a binary value and counts the number of times a feature occurs and the number of times it doesn't. In the multinomial model, the number of words is represented by a multinomial distribution, and it counts the frequency of the words in the document. In this research, a multinomial model is used.

Random forest

A random forest is a decision-making algorithm that creates a subset from data and uses a decision tree on each piece of data. The next step involves combining several decision trees and making a decision based on the majority [66]. The approach is suitable for problems involving regression and classification when the data set is large. The algorithm can handle a large dataset by providing a subset from them and then running a parallel decision tree on each subset, which makes it rather slow. The random forest has several advantages such as simplicity in implementation, flexibility in regression and classification problems,

reduced overfitting problems, which makes it a powerful machine learning algorithm [67]. Random forest algorithms combine several decision trees, resulting in a more accurate algorithm, but also a slower one. Each decision tree represents a sequence of decisions. Compared to the random forest, the decision tree algorithm is easier to visualize, quicker, and less accurate.

Logistic regression

Logistic regression is a linear classification algorithm in which the logistic function is used to model the dependent variable. This type of supervised learning uses labelled data for training, and it can use training data to describe new data and relationships between variables [68]. The logistic regression model does not make any assumptions about the distribution of data, and it can be learnt linear relationship from data. It is a fast, interpret and efficient algorithm that is used widely in binary and multiple-class classification. Logistic regression is using the logistic function or sigmoid function to transform predicted values into probabilities. A sigmoid function can be used in a probabilistic problem where the probability of anything happening is only between 0 and 1. A sigmoid function is an activation function that is used to convert the input to another value between the range 0 and 1.

Support vector machine

Support vector machine (SVM) is a supervised learning algorithm which is dividing the data into different classes by fitting the line or "hyperplane" among the samples. As shown in Figure 2-1, SVMs maximize the margin of their classifiers by using support vectors, which are points close to the hyperplane, as well as optimizing the hyperplane's location and orientation [69]. In SVM, the goal is to find a hyperplane that best separates the classes, and it can be applied to both classification and regression problems. The advantages of SVM are that it is accurate, reduces overfitting problems, works well in high-dimensional spaces, is memory efficient, and solves complex problems. On the other hand, it may require more time to process and may be more difficult to interpret compared to other machine learning algorithms. Linear SVC which is the implementation of SVM is used in this research. The idea of Linear SVC is dividing the data into

different classes by fitting the line or "hyperplane" among the samples, and it uses a kernel function to find the optimal separating hyperplane [69].

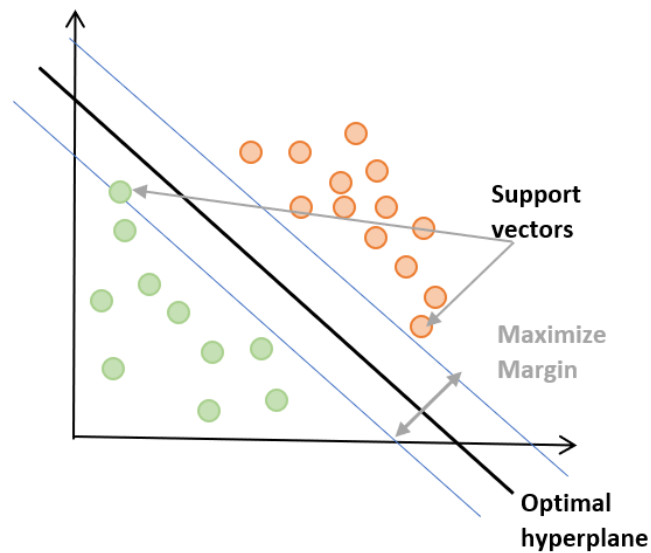


Figure 2-1. SVM for two-class classification

Neural network

Neural networks (NN) are machine learning networks that mimic the human nervous system [70]. NN can be used to model complex patterns by providing three components: input layer, hidden layer, and output layer. NN has usually supervised learning which required labelling; however, it can be trained on unlabeled data sets. NN can be utilized on classification and regression problems. There are three types of NN in machine learning: Artificial Neural Networks (ANN), Convolution Neural Networks (CNN), and Recurrent Neural Networks (RNN). In this research, different algorithms of CNN and RNN were applied for document classification.

Convolution Neural Networks (CNN)

CNN is a kind of NN that is using convolution operation as one of its layers. The convolution operation is sliding a filter or kernel on the input data, then it will produce a feature map[71]. The CNN needs to apply numerous convolutions to create different feature maps. The combination of all feature maps will be used as input of the next layer, which helps extract the right features from the input data. CNN can be used for image classification and recognition.

AlexNet

AlexNet is a GPU implementation of the CNN algorithm introduced by Alex Krizhevsky (2012). It has eight layers which include five convolutional layers and three fully connected layers. At the end of each layer, ReLU used as an activation function that can increase the learning rate, and prevent overfitting problems [72]. Using ReLU instead of traditional neuron models in CNNs is an important feature of the AlexNet. AlexNet is relying on the structure and layout of the documents to classify them. In this research, AlexNet was used for image-based document classification.

Faster R-CNN

Object detection is localizing and identifying an object or multiple objects in a single image which is an important part of image processing and image classification [73]. There are several object detection algorithms such as Fast R-CNN, Faster R-CNN, you only look once (YOLO), and single-shot detector (SSD).

Since accuracy is an important factor in this study, Faster R-CNN was selected as it has the highest accuracy compared to the others [31]. As a result, TensorFlow object detection API implementation of the faster R-CNN detector [74] was used for object detection. Faster R-CNN is a region proposal-based detector developed in 2015 by the Microsoft research team [34]. It has two stages: region proposal network (RPN)

and classifier. RPN will be used for generating region proposals and then a classification algorithm will be applied to regions to classify them into background or object classes.

Recurrent Neural Networks (RNN)

RNN is a repeating model of neural network that connects previous information to the present task through various loops. RNN is sending the output of a layer as feedback to the same layer. RNN is a useful model for sequence data and processing input of any length. However, the training procedures are slow and complex and when there is a gap between relevant information and output, the RNN is unable to connect the information. [75]. Some limitations of the RNN resulted in two modified versions of RNN: long short-term memory (LSTM) and gated recurrent unit (GRU) [76]. In this research, the LSTM algorithm was used for text-based document classification.

LSTM

A long short-term memory (LSTM) network is a type of RNN which is designed to fill the gap and avoid the long-term dependency problem. LSTM is using more functions and parameters to control the flow of information. LSTM network will take three decisions about the information: decide about the useless information which should be removed, decide about the new information which should pass to the next layer, and decide about the output of each layer [77]. The forget gate is using the sigmoid layer, the input gate is using the sigmoid layer and the tanh as an activation function and the output gate is using the sigmoid layer and the tanh as an activation function. The main advantage of LSTM is the ability to learn long-term dependencies and high prediction accuracy. It is suitable for time series, text generation, document classification, and sequential form problems. In this research, LSTM is used for text-based document classification.

2.5 Machine learning and deep learning algorithms

Machine learning and deep learning methods have been attractive topics in the last decade in all engineering disciplines. A growing set of new hardware for artificial intelligence (AI), availability of a large amount of data, and algorithm improvement led to an increase in the usage of deep learning methods. The application of deep learning is highly utilized in industries to solve a large number of complex problems such as image classification, object detection, and language processing. The field of construction engineering is well placed for applying deep learning methods. The architecture, engineering, and construction industry (AEC), like other engineering areas, relies on information technologies, and deep learning can provide lots of benefits. Literature review shows, a few researchers used machine learning and deep learning for information retrieval in the construction domain. Syeda-Mahmood [78] Introduced geometric hashing for extracting indexing keywords on engineering drawings. Soibelman et al. [79] applied statistical and machine learning algorithms for image classification and information retrieval in the construction industry. Brilakis et al. [80] used a kernel-based machine learning algorithm for retrieval information on construction image databases. Berkhahn et al. [55] used OCR and neural networks in AEC drawings to extract information about the dimensions of construction parts and inscription text. Also, a few researchers used machine learning and deep learning as vision-based techniques in AEC. Chi et al. [81] used a neural network with a multilayer for automated object identification on construction sites. Golparvar et al. [82] used machine learning algorithms for detecting workers and equipment. Memarzadeh et al. [83] used machine learning algorithms for automated 2D object detection in construction sites. Golparvar et al. [84] applied neural network algorithms for automated process monitoring of construction sites. Fang et al. [85] used a deep learning-based model for construction site image classification and detecting non-certified work in the site. Xu et al. [86] applied a deep convolutional neural network for damage identification of reinforced concrete columns from images.

Also, several researchers used machine learning algorithms for 3D model classification. Ip et al. [87] used surface curvature as a shape description and support vector machines (SVM) as a supervised machine

learning classifier. Huang and LeCun [88] presented a hybrid system for 3D model classification; CNN is used to learn feature description, and it was used as an input for SVM to do the classification. Qin et al. [89] designed an automated deep neural network classifier for 3D CAD models. Scheibel et al. [90] introduced a framework for dimensioning information extraction; they converted PDF files to HTML files and parsed HTML files to find all blocks and information.

Based on the literature review, it is found that the OCR technique, machine learning, and deep learning have been employed in the construction domain with acceptable results. However, their application in construction document classification is new. Although the number of applications of machine learning and deep learning is growing in different domains, there is little or no published information about employing them in the context of construction document classification. Based on the literature review, the following gaps were recognized:

- lack of appropriate tools for classification of scanned documents, which have limited text;
- lack of automated text-based approach for information extraction from scanned construction documents;
- lack of a domain-specific model for drawing and non-drawing classification; and
- lack of a classification model that includes all types of construction documents.

Chapter 3 Research methodology

To achieve the research objectives, three main tasks, as illustrated in Figure 3-1 were defined as data collection and analysis, design and implementation, and test and evaluation.

The data collection and analysis tasks focus on data preparation, literature review, classification model analysis, and selection. Based on this analysis, different classification methods were selected. The design and implementation task is focused on developing and executing established processes which have three main phases. Phase 1 involves developing classification based on text, which is focused on text extraction. After pre-processing steps to prepare the documents, the document's text will be extracted by the optical character recognition (OCR) engine and then various methods are used for document classification such as term frequency-inverse document frequency (TF-IDF) and text similarity approach, pre-defined keywords, and machine learning algorithms. Phase 2 involves developing classification based on construction document images. For this purpose, two different machine learning and deep learning algorithms were tested: TensorFlow object detection API and AlexNet. Phase 3 is developing title block detection and information extraction using TensorFlow object detection API to find the location of title block detection, and then the text of the title block was extracted by the OCR engine. The regular expression was applied on extracted text, and information such as revision number, drawing name, and numbers were extracted. In the test section, all the proposed models were assessed to select the best model based on their performance. Finally, the evaluation section was applied the chosen model to the second data set to determine the reusability of the model. As a result of the proposed framework, comprehensive automated document classification, automated title block detection and information extraction is expected.

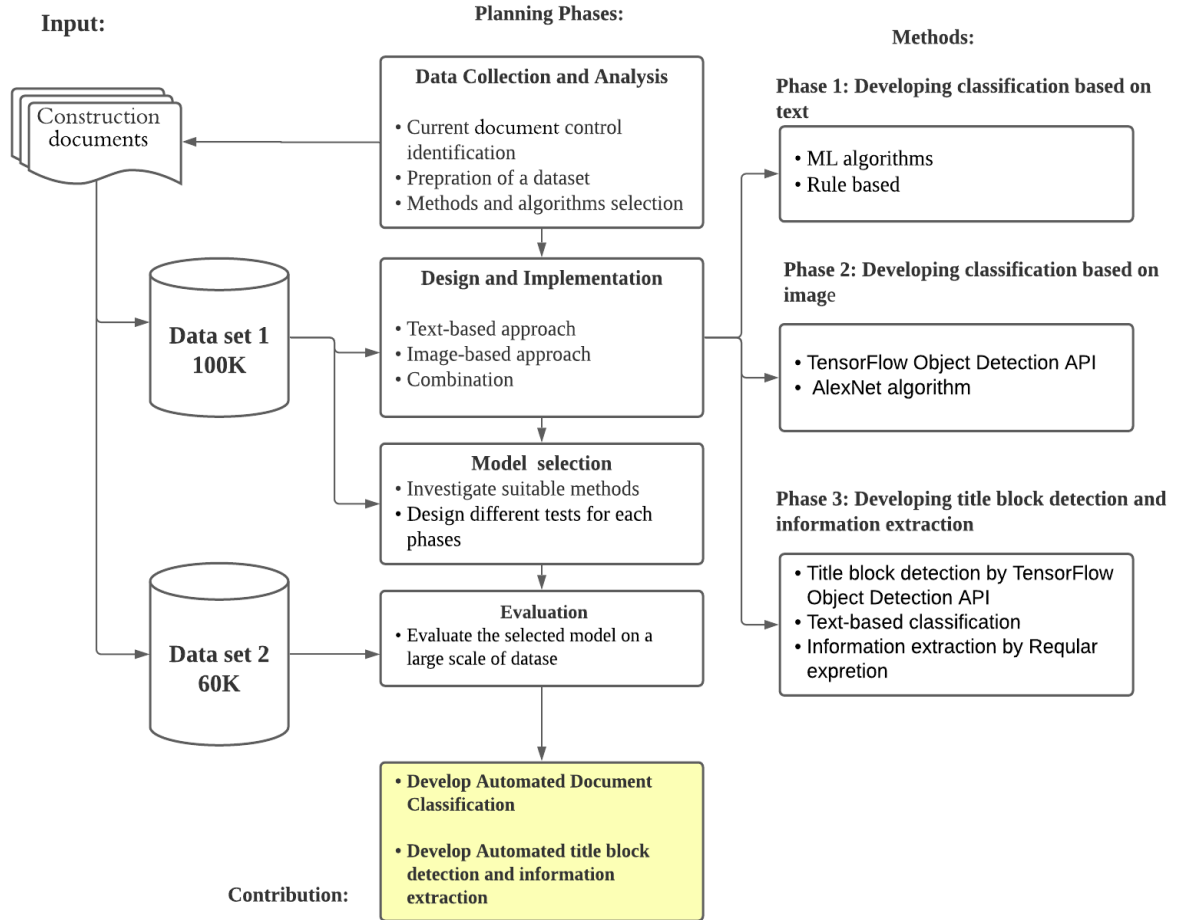


Figure 3-1. Overall research methodology

Chapter 4 Data collection and analysis

Data collection and analysis were defined as the first step of research methodology, which is the procedure of collecting a large amount of data, analyzing the current solution, identifying the possible solutions and selecting methods and algorithms. Under this task, the following data collection and research activities were completed:

1. Understand the current document control practice of the construction company and how documents are currently managed within the company. This task includes information gathering about different types and formats of construction documents, information flow, and their relation.
2. Identify the different types of documents that need to be retrieved and for what purposes, and the current process used for document retrieval. This step will also include developing a set of performance metrics for benchmarking future automated solutions against current practices.
3. Study the current manual classification process that is performed by document control teams and extract the critical document feature set that is utilized by the teams in this manual process. This activity will also include the preparation of a dataset that consists of a group of documents, a key feature set for each document, and one or more classification types identified and assigned to the document as part of the manual document control and classification process.
4. Once a dataset is prepared, alternative supervised learning classification algorithms will be evaluated to assess their performance in recognizing and assigning document classes.
5. Selection of classification methods and algorithms will be carried out by assessing their accuracy in retrieving relevant documents compared to the current retrieval process currently utilized in the company.

4.1 Data set preparation

Data were collected for case study analysis including 100,000 documents that belong to 15 construction projects with 32 different document types. Figure 4-1 shows 32 document types and the number of documents for each data type. The majority of construction documents belong to isometric drawings with

37,967 documents, datasheet with 8893, and wiring diagram with 8496 documents. Table 4-1 shows the percentage of 32 document types, of which 19 belong to engineering drawings, and 13 belong to non-drawing documents. Table 4-2 illustrates the document type and percentage of each group. While 81.35% of documents are engineering, 18.65% are non-engineering. Figure 4-2 displays samples of drawing documents such as area classification, block diagram, logical diagram, and isometric diagram. Figure 4-3 shows an example of non-drawing documents such as material take-off, work package, datasheet, and manual. Most of the documents have more than one layout with different locations of the title block. Figure 4-4 shows two different formats of the bill of materials document. The title block is located at the top of the first layout, located at the bottom of the second layout. In addition, the provided data set includes 60000 documents used for test and evaluation purposes in Chapter 7. All documents were in PDF format, which should be converted to image format to use OCR text detection. Adobe Acrobat Pro DC software was used to convert PDFs to PNGs. Since the information that needs to extract is usually on the first page of each document, only the first page of each document was selected.

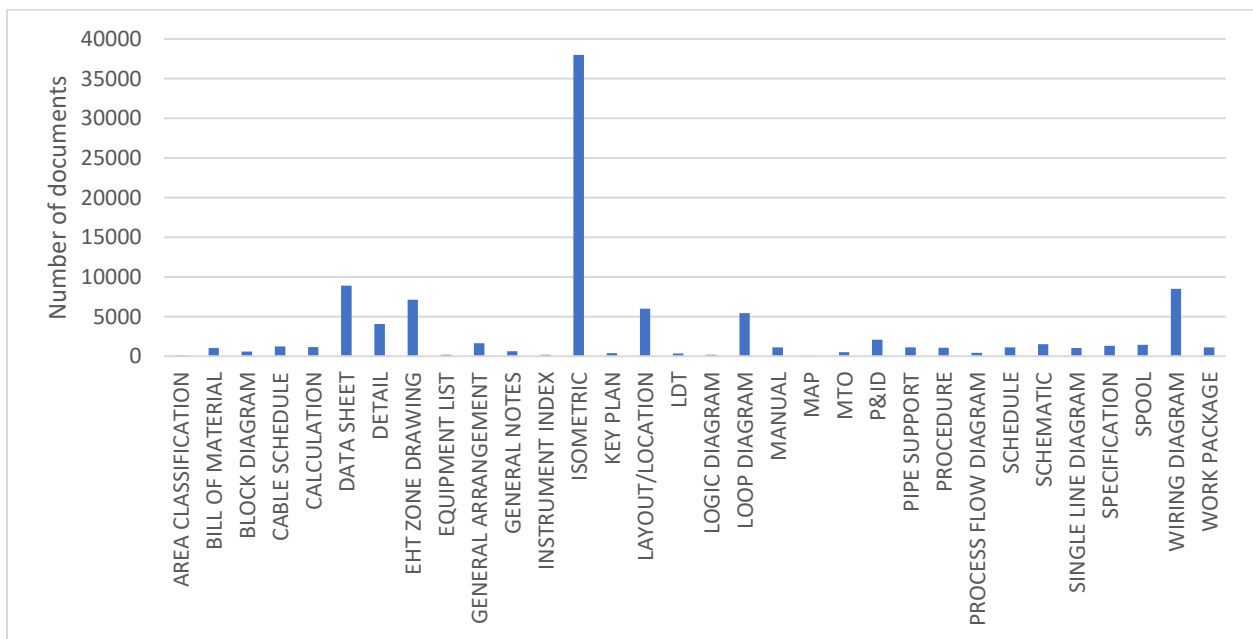


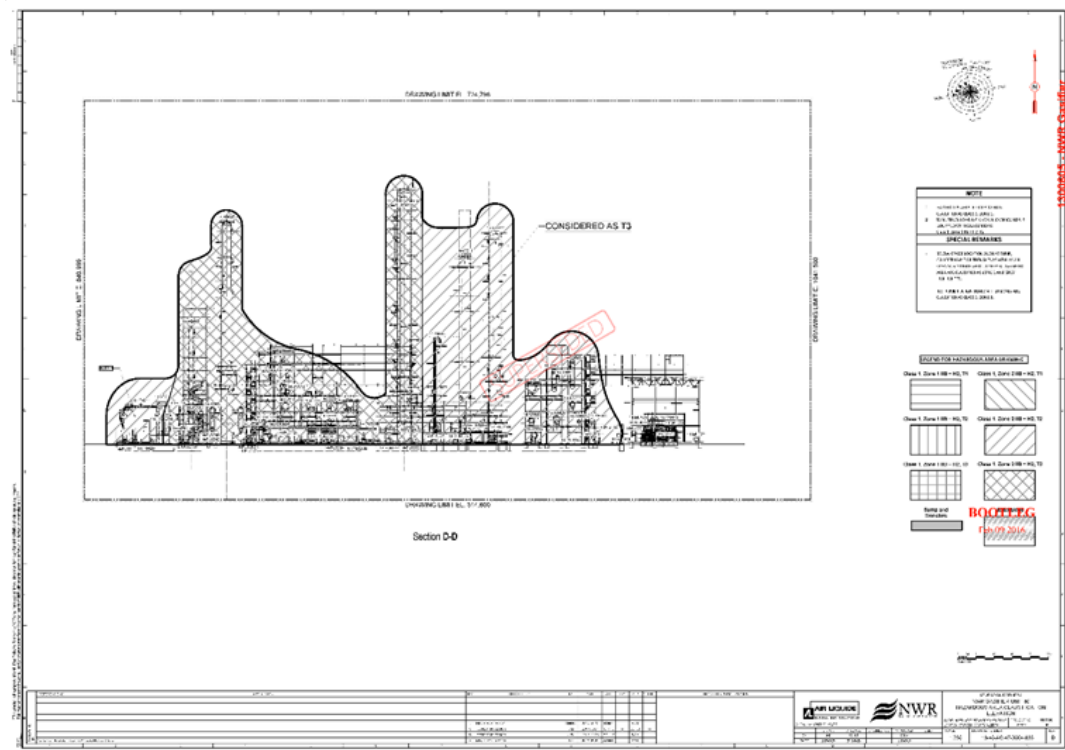
Figure 4-1. Document types in the first dataset

Table 4-1. Percentage of document types

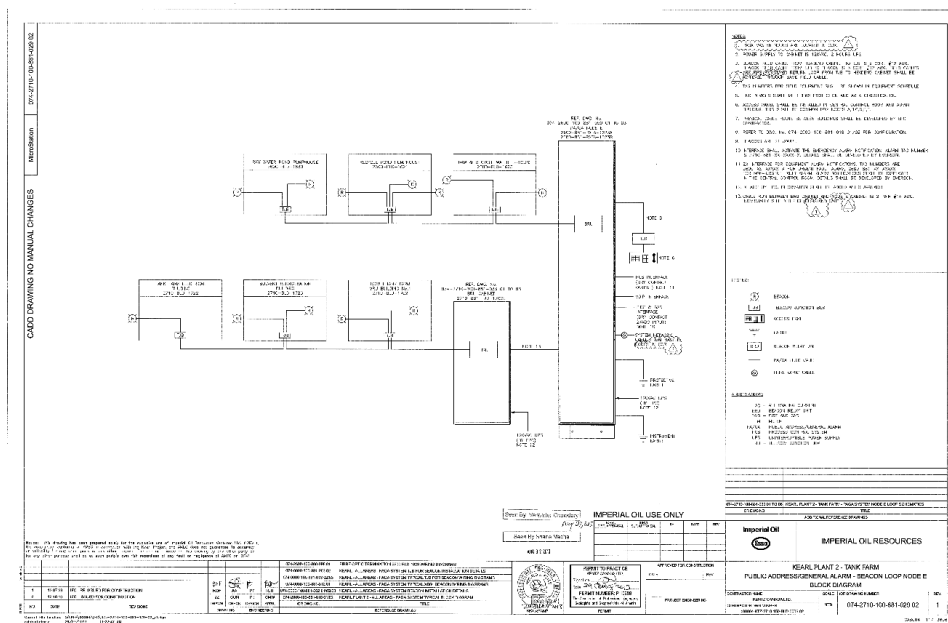
Document Type	% Documents
AREA CLASSIFICATION	0.12%
BILL OF MATERIAL	1.03%
BLOCK DIAGRAM	0.59%
CABLE SCHEDULE	1.24%
CALCULATION	1.18%
DATA SHEET	8.94%
DETAIL	4.10%
EHT ZONE DRAWING	7.17%
EQUIPMENT LIST	0.20%
GENERAL ARRANGEMENT	1.65%
GENERAL NOTES	0.62%
INSTRUMENT INDEX	0.18%
ISOMETRIC	38.17%
KEY PLAN	0.37%
LAYOUT/LOCATION	6.03%
LDT	0.35%
LOGIC DIAGRAM	0.20%
LOOP DIAGRAM	5.47%
MANUAL	1.13%
MAP	0.02%
MTO	0.50%
P&ID	2.08%
PIPE SUPPORT	1.13%
PROCEDURE	1.07%
PROCESS FLOW DIAGRAM	0.42%
SCHEDULE	1.10%
SCHEMATIC	1.51%
SINGLE LINE DIAGRAM	1.04%
SPECIFICATION	1.30%
SPOOL	1.44%
WIRING DIAGRAM	8.54%
WORK PACKAGE	1.11%
Total	100.00%

Table 4-2. Percentage of drawing and non-drawing documents

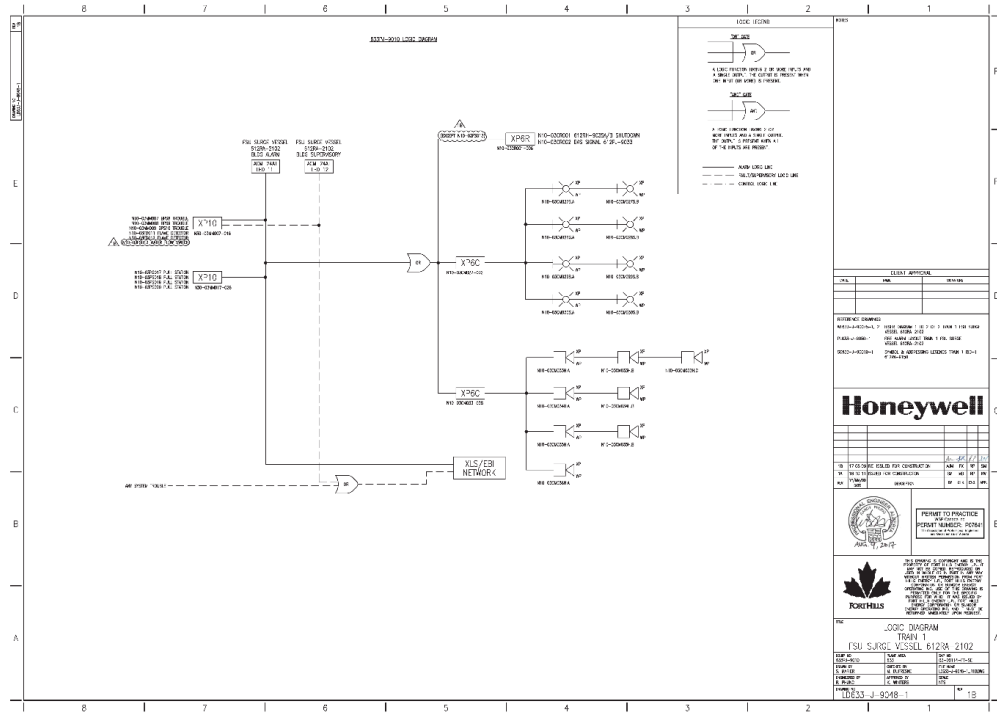
Drawing document type	% Documents	Non-Drawing document type	% Documents
AREA CLASSIFICATION	0.12%	BILL OF MATERIAL	1.03%
BLOCK DIAGRAM	0.59%	CABLE SCHEDULE	1.24%
DETAIL	4.10%	CALCULATION	1.18%
EHT ZONE DRAWING	7.17%	DATASHEET	8.94%
GENERAL ARRANGEMENT	1.65%	EQUIPMENT LIST	0.20%
ISOMETRIC	38.17%	GENERAL NOTES	0.62%
KEY PLAN	0.37%	INSTRUMENT INDEX	0.18%
LAYOUT/LOCATION	6.03%	LDT	0.35%
LOGIC DIAGRAM	0.20%	MANUAL	1.13%
LOOP DIAGRAM	5.47%	MTO	0.50%
MAP	0.02%	PROCEDURE	1.07%
P&ID	2.08%	SCHEDULE	1.10%
PIPE SUPPORT	1.13%	WORK PACKAGE	1.11%
PROCESS FLOW DIAGRAM	0.42%		
SCHEMATIC	1.51%		
SINGLE LINE DIAGRAM	1.04%		
SPECIFICATION	1.30%		
SPOOL	1.44%		
WIRING DIAGRAM	8.54%		
Total	81.35%		18.65%



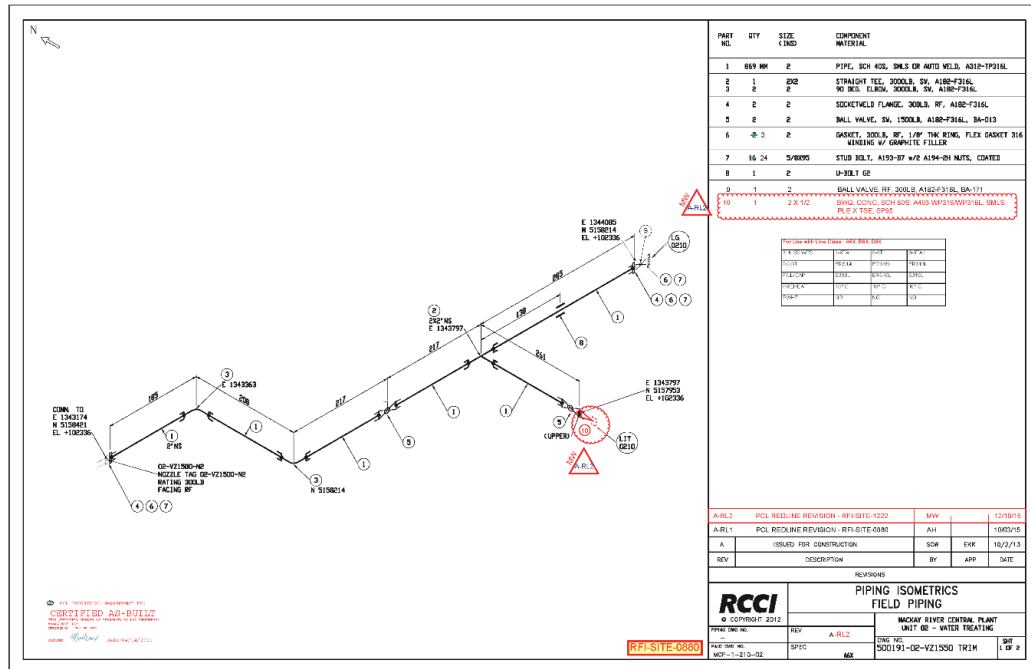
a) Area classification



b) Block Diagram



c) Logical Diagram



d) Isometric Drawing


Figure 4-2. Sample of drawing documents

Item	Area	Insulator Type	Equip. No.	Equipment Name	IPD Item	AVI Drawing #	Vendor	Operating Temperature (°C)	Storage Temperature (°C)	WCC	Thickness (mm)	Depth (mm)	Depth at Lip (mm)	Estimated Surface Area (sq. m)	
Material Take-Off															
INSULATION FOR CHUTES in Area 162-163-167-172-476 - 180-190				Client: AGRUM	Document No: 0007030ET	35496-000-001-7030	Prepared by: L. Hafard	Date: 2015/11/27							
				Project: Project VAULT	Project No: 334952	Revision: 01	Reviewed by: M. Avil	Date: 2013/11/27							
				Capacity: 2.8 MW/year			Approved by: C. Samothedra	Date: 15-11-17							
AREA 162 CHUTES															
1	01	162	PP	02864	Product Drying Chute	180A7341	182D2731	JNE	190	---	CARBON STEEL	---	---	4.7	
AREA 163 CHUTES															
1	01	163	HC	02540	Special Standard Screw Conveyor (15 811) Discharge Chute	100A7370	182D2720	JNE	190	---	CARBON STEEL	---	---	9.3	
2	01	163	HC	02541	Spiral Mixer (27 824) to CPES Mixing Screw Conveyor (15 825) Chute	100A7370	182D2738	JNE	190	---	CARBON STEEL	---	---	1.6	
3	01	163	HC	02542	Spiral Mixer (27 824) to CPES/CPM Mixing Screw Conveyor (15 822) Chute	100A7370	182D2739	JNE	190	---	CARBON STEEL	---	---	2.0	
4	01	163	HC	02544	Special Standard Screw Conveyor (15 823) Discharge Chute	100A7370	182D2712	JNE	190	---	CARBON STEEL	---	---	1.8	
5	01	163	HC	02591	Special Standard Bucket Elevator Discharge Chute	100A7370	182D2713	JNE	190	---	CARBON STEEL	---	---	9.0	
6	01	163	HC	02567	Special Standard Screw Conveyor (15 823) Feed Chute	100A7370	182D2712	JNE	190	---	CARBON STEEL	---	---	10.6	
7	01	163	HC	02815	Wall Product Screen Feed Chute Conveyor (15 829) to CPES/CPM Mixing Screw Conveyor	100A7370	182D2710	JNE	190	---	CARBON STEEL	---	---	7.2	
8	01	163	HC	02818	CPES/CPM Mixing Screw Conveyor (15 800) to CPES/CPM Transfer Chute Conveyor (15 705) Chute	100A7370	182D2702	JNE	190	---	CARBON STEEL	---	---	8.6	
9	01	163	HC	02817	Wall Product Screen Feed Chute Conveyor (15 828) to CPES/CPM Transfer Chute Conveyor (15 245) Chute	100A7370	182D2703	JNE	190	---	CARBON STEEL	---	---	8.1	
AREA 167 CHUTES															
1	01	167	PP	02962	CPES Mixing Sample Feed Chute	100A7402	182D2702	CFI	190	---	CARBON STEEL	---	---	3.4	
2	01	167	PP	02920	CPES Inlet Chute Conveyor Discharge Chute	100A7402	182D2703	CFI	190	---	CARBON STEEL	---	---	7.7	
3	01	167	PP	02923	CPES Mixing Sample Discharge Chute	100A7402	182D2702	CFI	190	---	CARBON STEEL	---	---	7.7	
4	01	167	PP	02864	Inlet Chute	100A7402	182D2702	CFI	190	---	CARBON STEEL	---	---	1.4	
AREA 172 CHUTES															
1	01	172	HC	02820	Position Upper Transfer Chute Conveyor (16 789) to Screening Spiral Transfer Chute Conveyor (16 792) Chute Detail	100A7261	172D2716	CFI	190.0	---	CARBON STEEL	---	---	0.9	
2	01	172	PP	02781	Stackback Dust Screw Feeder (15 713) Discharge Chute Detail	100A7261	172D2722	CFI	20 - 190	---	CARBON STEEL	---	---	6.6	
3	01	172	HC	02820	Position Upper (25 710) Discharge Chute Detail	100A7261	172D2716	CFI	190.0	---	CARBON STEEL	---	---	1.4	
4	01	172	HC	02821	Position Upper Discharge Screw Conveyor (15 709) Discharge Chute Detail	100A7261	172D2716	CFI	190.0	---	CARBON STEEL	---	---	13.6	
5	01	172	HC	02822	Position Upper Cyclone Discharge Conveyor (22 702) to Position Upper Cyclone UHF Screw Conveyor (16 703) Chute Detail	100A7261	172D2717	CFI	190.0	---	CARBON STEEL	---	---	15.1	
6	01	172	HC	02823	Position Upper Cyclone UHF Screw Conveyor (15 703) Chute Detail	100A7261	172D2717	CFI	190.0	---	CARBON STEEL	---	---	14.4	
7	01	172	HC	02824	Position Upper Discharge Screw Elevator (21 710) Discharge Chute Detail	100A7261	172D2718	CFI	190 - 210	---	CARBON STEEL	---	---	15.8	
8	01	172	HC	02825	Position Upper Transfer Chute Conveyor (16 789) to Position Upper Transfer Chute Conveyor (16 792) Chute Detail	100A7261	172D2716	CFI	190 - 210	---	CARBON STEEL	---	---	7.7	
9	01	172	HC	02826	Position Upper Transfer Chute Conveyor (16 792) to Position Upper Transfer Chute Conveyor (16 792) Chute Detail	100A7261	172D2716	CFI	190.0	---	CARBON STEEL	---	---	29.5	
10	01	172	PP	02809	Position Upper Feed Chute Conveyor (16 725) Discharge Chute Detail	100A7261	172D2717	CFI	20.7	---	316 SS	---	---	3.3	
11	01	172	HC	02810	Screening Spiral Transfer Chute Conveyor (16 789) Discharge Chute Detail	100A7261	172D2709	CFI	190	---	CARBON STEEL	---	---	41.7	

a) MTO (Material Take-off)


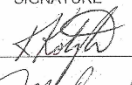



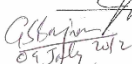
GENERAL	1	Project Name	MCRP	Tag No.	MCF-AA-PSV-3006	
	2	Plant Name	MCF	PSV No.	MCF-PS-210A-0601-02	
	3	Area Name	Pads	Service	MCF-AA-E000 PAD RG RTR RG FROM PAD EDGE RELIEF	
	4	Unit Name	AA	Location	FIELD	
	5	Arm. Temp.	-40 to 38 °C	Line No.	Z-C11-RG-26037	
	6	Arm. Press.	95.6 kPa-a	Equipment No.	MCF-AA-E4000	
	7	Service	Low Temperature Sweet Hydrocarbon	Area Class.	Class I, Zone 2, Grp. IIA, T2A	
	8	Design Pressure	9930 kPa-g			
	9	Design Temp.	38 °C			
	10					
SERVICE CONDITIONS	11	Fluid	Fluid Phase	Residue Gas	Gss/Vapor	
	12	Relief Capacity & Units		193.2	kg/h	
	13	Molecular Weight	Operating S.G.	16.4	0.587	
	14	Specific Heat Ratio		1.249		
	15	Compressibility Factor		1		
	16	Viscosity at Relieving Temperature		0.02	cP	
	17	Operating Pressure	Set Pressure	9930	kPa-g	
	18	Operating Temperature	Relief Temperature	5	279 °C	
	19	Percentage Overpressure		21	%	
	20	Constant Back Pressure		0	kPa-g	
ORIFICE	21	Variable Back Pressure		100	kPa-g	
	22					
	23	Capacity Calculation Basis		Fire		
	24	Orifice Area Required		0.006	in²	
	25	Orifice Area Selected	Designation	0.11	in²	
	26	ASME			D	
	27	Selected Valve ASME Capacity		API - 3753.52	kg/h	
	28	Actual Valve Capacity				
	29					
	VALVE	30	Valve Classification		Safety Relief	
31		Valve Design Type		Conventional		
32		Valve Action		Pop		
33		Code Stamp		ASME Section VIII		
34		Cap	Lever	Screwed	None	
35						
36		Body Material		A352 LCC		
37		Inlet Process Connection	Size	1.5	m	
38			Type	RF Flange	ANSI 900	
39		Outlet Process Connection	Size	2	m	
BODY & BONNET	40		Type	RF Flange	ANSI 300	
	41	Bonnet Material		A352 LCC		
	42	Bonnet Type		Closed Type		
	43	Studs		ASME SA 193 B7M		
	44	Nuts		ASME SA 194 2HM		
	45					
	46	Spring Material	Spring Seal	Alloy Steel		
	47	Nozzle Type	Nozzle Material	Full	316 SST	
	48	Seat and Disc				
	49	Main Valve Soft Goods	Pilot Soft Goods			
TRIM	50	Pilot	Line			
	51	Pilot Body	Pilot Trm			
	52					
	53	Backflow Preventer	Test Gag			
	54	Auxiliary Filter				
	55	NACE MR-0175		N/A		
	56					
	57					
	58					
	59					
OPTIONS	60	Hydro. Test		Yes		
	61	Other				
	62	NDE		Hardness Testing		
	63	Manufacturer		Consolidated		
	64	Model No.		1914-000-2-C1-MS-31-RF-GS-HP-SPEC		
	65	P. O. No.	Item No.	MCF-POS-PR-00037		
	66	Serial No.				
	67	Alberta CRN		065450 52		
	68					
	69					
NOTES	70					
	71					
	72					
	73					
	74					
	75					
	76					
	77					
	78					
	79					
INSTRUMENT DATA SHEET						
1 PHASE PRESSURE RELIEF VALVE						
DATA SHEET DOC NO: DS-MCF-AA-PSV-3006						
SLI NO:						
SHEET: 1 OF 2						
CODE: 161 REV: 0						
REV	DATE	DESCRIPTION	BY	PROJ. CH.	INTL. USE	APP. USE
0	09/15/2014	Issued for fabrication	EA	GS	CG	RD UK

b) Datasheet

 SNC-Lavalin / PCL Joint Venture	Engineering Work Package Underground Conveyor Structural Mainline Pre-Assembly	Revision		Page
		#	Date	
Document No.	110B7009EW	03	2012-07-06	1
EWP No.	10-110b-ST-001	334562-0110-43EW-7009		

CLIENT: AGRUM

PROJECT: VAULT PROGRAMME

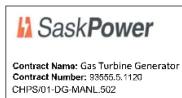
	SIGNATURE	DATE
PREPARED BY:  K. Kolyk		2012/07/06
REVIEWED BY: J. Rempel		2012-07-03
APPROVED BY:  B. De Jager		2012-07-03
		2012-07-09

ISSUE/REVISION INDEX

Issue Code	Revision					Revision Details
	No.	By	Rev'd.	App.	Date	
RC	00	KK	JR	BDJ	2011-11-25	Released for Construction
RC	01	KK	JR	BDJ	2012-02-03	Released for Construction
RC	02	KK	JR	BDJ	2012-05-30	Released for Construction
RC	03	KK	JR	BDJ	2012-07-06	Released for Construction

Issue Codes: RC = Released for Construction, RD = Released for Design, RF = Released for Fabrication, RI = Released for Information, RP = Released for Purchase, RQ = Released for Quotation, RR = Released for Review and Comments.

c) Work package



Pentair Valves and Controls US LP
605 Territorial Drive, Unit B
Bolingbrook, IL 60440
630-343-3333

TITLE:	Fuel Gas Emergency Stop Valve
DOCUMENT:	Installation, Operation & Maintenance Manual
CUSTOMER:	Siemens Energy, Inc.
PROJECT NAME:	CA1054 Saskpower Chinook Power Station
SIEMENS PO#:	4500764874
UNID:	508600450
REVISION:	0

Table of Contents:

- IOM Series 30,000 Zero Leakage Valve
- IOM Morin Pneumatic Actuator Models B,C, & S – OMI1013
- IOM Asco 3 way solenoid valve – V 6928 R4
- IOM Bellofram Type 50 Regulators – 214-541-000-030 Rev C
- IOM Deltrol Quick Exhaust – TVGCSSD2694 Rev 4
- IOM Westlock AccuTrak Rotary Limit Switch – Tech-386_r0.0 D.W.O 17012

Manual


Figure 4-3. Sample of non-drawing documents

REV No.	ITEM No.	UNIT	QTY	DESCRIPTION	UNIT PRICE (\$)	TOTAL PRICE (\$)
1				BUILDING EXPANSION FOOTPRINT FLOOR @ EL. 1654'-0" TRUCK LOADING ASBLE EXPANSION WITH CONTROL ROOM 60'-0" x 36'-6" + 40'-0" x 34'-0" = 3550 sf FLOOR @ EL. 1679'-0" NEW BLOWBACK TANK ROOM 12'-0" x 30'-0" = 360 sf FLOOR @ EL. 1679'-0" TRUCK LOADOUT CONVEYOR ROOM EXPANSION & NEW COMPRESSOR ROOM 60'-0" x 36'-6" + 40'-0" x 34'-0" = 3550 sf FLOOR @ EL. 1660'-0" NEW CABLE ROOM 40'-0" x 36'-6" = 1480 sf FLOOR @ EL. 1702'-10" NEW MODULAR ELECTRICAL SWITCH ROOM #10 36'-6" x 40'-0" = 1480 sf		
2	4145	sf	4145	METALS 1 1/2" GALVANIZED STEEL ROOF DECK (8 MIL BARRIER SYSTEM COATING WITH STAINLESS STEEL FASTENERS)		
3	4145	sf	4145	THERMAL AND MOISTURE PROTECTION ROOF SYSTEM TYPE 1 - ELASTOMERIC COATING, TWO COMPONENT POLYURETHANE SPRAY ON 1" FOAM INSULATION (R-20, 5/8" DENS DECK)		
4	436	L.F.	436	14" HIGH ROOF PARAPET INCLUDING WOOD BLOCKING, STAINLESS STEEL UPSTAND, INSULATION, SPRAY APPLIED ELASTOMERIC COATING & STAINLESS STEEL CAP FLASHING		
5	14354	sf	14354	WALL CLADDING PRE-FINISHED 1 3/4" FRP PANEL ON STRUCTURAL GIRTS		
6	1087	sf	1087	PRE-FINISHED SELF FRAMING WALL PANEL		
7	8		8	DOORS AND WINDOWS EXTERIOR FRP DOORS WITH FRP FRAMES 3'-0"x7'-0"x1 3/4" & HARDWARE		
8	3		3	INTERIOR FRP DOORS WITH FRP FRAMES 3'-0"x7'-0"x1 3/4" & HARDWARE		
9	2		2	EXTERIOR RUBBER ROLL UP DOOR 14'-6" x 17'-0" (POWER OPERATED)		
10	3		3	INTERIOR WINDOWS 1-17'-4" x 4'-0" (EQ SPACED IN 4 BAYS), 1-4'-8"x4'-0", 1-4'-10"x4'-0" PRESSED FRAME WITH DOUBLE GLAZING		
11	5510	sf	5510	FINISHES CONCRETE FLOOR - CORROSION RESISTANCE COATING & SEALER/HARDENER		
12	390	sf	390	ONE HOUR SHAFT WALL ULC W452 SYSTEM		

a) layout 1

NOTE:
THIS CHART IS TO BE READ IN CONJUNCTION WITH DRAWING 172D8520.

Association of Professional Engineers & Geoscientists of Saskatchewan
CERTIFICATE OF AUTHORIZATION
SNC-Lavalin Inc.
Number C0834
Permission to Consult held by:
Discipline: Mech. Sr. Reg. No. 26715
Signature: [Signature]



ITEM #	SYSTEM #	DRIP TRAY TAG NUMBER	AREA	SERVICE	PID	CONTROL STATION LINE NUMBER	DRIP TRAY DRAIN LINE	LENGTH (ft.-in.)	HEIGHT (ft.-in.)
1	6	96751	171	DIS	100J7767	2"-DIS-PSC2XR-164-0006	2"-DIS-PPG2BD-171-0701	8'-6"	2'
2	14	96744	172	DP	100J7805	1 1/2"-DP-PSC2XR-172-0014	2"-DP-PPG2BR-172-0708	9'	4'
3	6	96750	171	DP	100J7757	3"-DP-PSC2XR-171-0006	2"-DP-PPG2BR-171-0709	11'	2'
4	7	96752	171	DP	100J7779	1 1/2"-DP-PSC2XR-171-0001	2"-DP-PPG2BR-171-0710	8'-2"	2'-8"
5	7	96753	171	DP	100J7780	1 1/2"-DP-PSC2XR-171-0002	2"-DP-PPG2BR-171-0711	7'	2'-8"
6	9	96754	171	DP	100J7781	1 1/2"-DP-PSC2XR-171-0003	2"-DP-PPG2BR-171-0712	7'	2'-8"
7	11	96758	172	DP	100J7800	2"-DP-PSC2XR-172-0004	2"-DP-PPG2BR-171-0713	9'-8"	2'-8"
8	19	96768	172	DP	100J7850	3"-DP-PSC2XR-172-0005	2"-DP-PPG2BR-172-0714	3'	2'
9	19	96770	172	DP	100J7850	3"-DP-PSC2XR-172-0005	2"-DP-PPG2BR-172-0715	7'	2'
10	12	96741	172	FLO	100J7785	1"-FLO-PCC2KR1-172-0004	2"-DF-PPG2BD-172-0014	4'-2"	2'-9"
11	20	96756	172	FLO	100J7783	1"-FLO-PCC2KR1-171-0006	2"-FLO-PCC2KR1-172-0701	4'-2"	2'-4"
12	11	96759	172	FLO	100J7800	1"-FLO-PCC2KR1-172-0005	2"-FLO-PCC2KR1-172-0702	6'	2'-8"
13	14	96764	172	FLO	100J7808	1"-FLO-PCC2KR1-172-0008	2"-FLO-PCC2KR1-172-0703	6'	3'-3"
14	16	96769	172	FLO	100J7817	1"-FLO-PCC2KR1-172-0009	2"-FLO-PCC2KR1-172-0705	5'	3'
15	19	96767	172	FLO	100J7850	1"-FLO-PCC2KR1-172-0010	2"-FLO-PCC2KR1-172-0704	5'	2'-8"
16	14	96745	172	FR	100J7808	1/2"-FR-PSC2XR-172-0007	2"-DF-PPG2BD-172-0015	3'-4"	3'-4"
17	14	96747	172	FR	100J7817	1/2"-FR-PSC2XR-172-0012	2"-DF-PPG2BD-172-0011	3'-4"	3'-0"
18	19	96749	172	FR	100J7851	1/2"-FR-PSC2XR-172-0003	2"-FR-PSC2XR-172-0022	3'-3"	2'-10"
19	12	96757	172	FR	100J7786	1/2"-FR-PSC2XR-172-0001	2"-FR-PPG2BD-172-0700	5'	3'
20	11	96760	172	FR	100J7801	1/2"-FR-PSC2XR-172-0011	2"-FR-PPG2BD-172-0701	3'-3"	2'-10"
21	11	96761	172	FR	100J7802	1/2"-FR-PSC2XR-172-0002	2"-FR-PPG2BD-172-0702	4'-6"	3'
22	11	96762	172	FR	100J7802	1/2"-FR-PSC2XR-172-0005	2"-FR-PPG2BD-172-0703	4'-8"	3'-6"
23	16	96765	172	FR	100J7810	1/2"-FR-PSC2XR-172-0020	2"-FR-PPG2BD-172-0704	3'-6"	3'

JV Vault
SNC-Lavalin / PCL Joint Venture

Agrium

PROJECT VAULT
334562-0100-44ET-7750 Rev 01
192B7256ET
Prep'd: ww Date: 2013-05-24
Checked: PS Date: 12-10-2017
App'vd: [Signature]

b) Layout 2

Figure 4-4. Bill of materials layouts

4.2 Pre-processing steps for improving image quality

The data set includes different sizes of documents with different qualities. Based on the literature review[91–94], the resolution of images should be at least 300 Dots Per Inch (DPI) for a better text detection result. The first step is to resize all the documents which will affect the resolution of images. In the second step, Matlab’s image processing toolbox is used for improving the quality of images. It has different functions and filters that can be applied to modify the images. `rgb2gray` filter is used to convert documents to grayscale images. `Imadjust` filter is applied to increase the contrast and brightness of the output image, and the Median filter is used to remove the noise from the grayscale pictures [95]. Figure 4-5 shows an original image and the enhanced images using Matlab’s image processing toolbox.

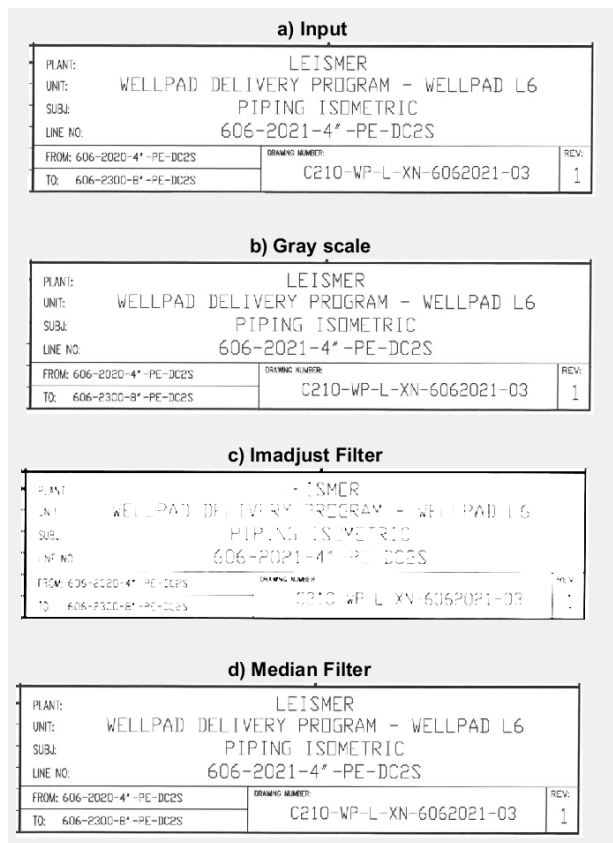


Figure 4-5. Steps of image enhancement

4.3 Layout analysis

Layout analysis such as page segmentation and region classification have a vital role in text detection. Different regions such as texts, images, and tables should be identified to extract the text correctly. Layout analysis defines the possible location of the text that needs to be extracted, increasing the OCR accuracy and extracting more useful text from each document [96]. As Figure 4-6 shows, the dataset has different layouts, which need to classify into text and non-text segmentation. The location of the table of information is variable, as indicated by the 'A' in Figure 4-6.

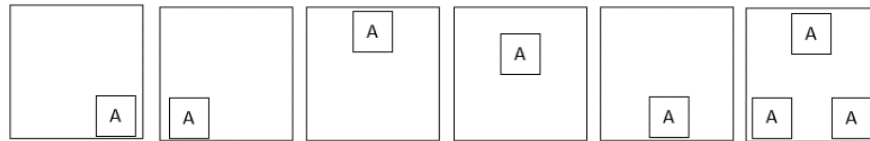


Figure 4-6. Possible locations of the table of information

Connected component analysis or connected component labelling is the basic image processing algorithm. The connected component analysis searches all un-labelled pixels and groups pixels that belong to the same connected component or object [97]. For the case study, connected component analysis was used to classify text and non-text segmentation. Non-text segmentation involves tables, images, and lines. In the next step, all the text segmentations will be processed through Matlab's OCR in Computer Vision System Toolbox™.

Chapter 5 Design and implementation

Design and implementation were defined as the second step of research methodology, which has three primary phases to meet the research objectives progressively.

Phase 1. Developing classification based on text

Phase 2. Developing classification based on image

Phase 3. Developing title block detection and information extraction

5.1 Phase 1: Developing classification based on text

Developing classification based on the text was designed in five stages. Figure 5-1 indicates the five stages of Phase 1. The first stage includes the OCR technique and text extraction. The second stage includes term frequency-inverse document frequency (TF-IDF) methods and analysis of four different classification algorithms. The third stage contains pre-defined classification. The fourth stage is focused on long short-term memory (LSTM) classification, and the fifth stage is comparing the results of previous steps. The following section will discuss the stages in detail.

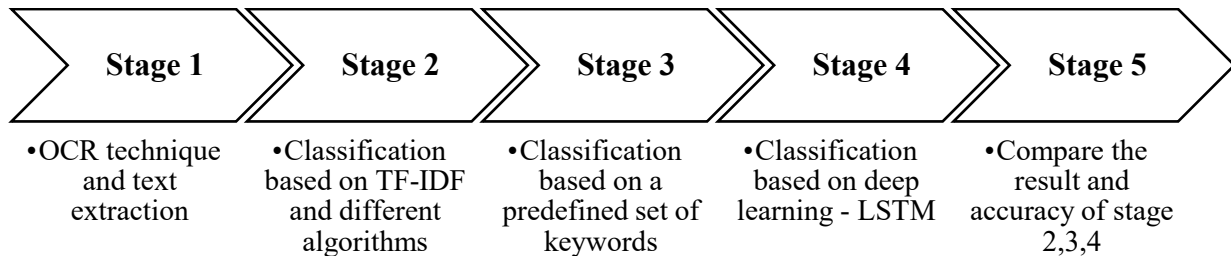


Figure 5-1. Proposed method of construction document classification based on text

5.1.1 OCR technique and text extraction

Different OCR engines were tested for text detection. Matlab OCR was used, which includes the following steps: Depending on whether resizing is necessary, start by resizing the image. Later OCR engine will apply to documents, and it is not working well on small text sizes. Based on Matlab recommendation, the height of a lowercase “x”, or comparable character in the input image, should be greater than 20 pixels. If the OCR engine did not work well due to the small text size, the size of the image should be increased. In the next step, the input image should convert to a binary image to reduce algorithm complexity and be used for connected component analysis. A binary image will convert the original image into a black and white image. It is also a valuable method to remove noise when a document has a dark background. The function of `BW = imbinarize (I)` creates a binary image from 2-D or 3-D grayscale image I. Furthermore, the connected component analysis uses the binary image as an input which is a set of pixels. Then it will search all pixels to find what pixels belong to the same region. As a result, each pixel will be labelled as a background pixel or object. The number of objects and the pixel index lists are the output of the connected component. The function of `CC = bwconncomp (BW)` returns the connected components CC found in the binary image BW. Figure 5-2 illustrates the process of the connected component analysis. And Figure 5-3 shows the sample code of connected component analysis.

Revision Details

Issued for Construction

a) Original Image

Revision Details

Issued for Construction

b) Binarize Image

z = 251x833 logical array

	1	2	3	4	5	6	7	8	9	10
229	1	1	1	1	1	1	1	1	1	1
230	1	1	1	1	1	1	1	1	1	1
231	1	1	1	1	1	1	1	1	1	1
232	1	1	1	1	1	1	1	1	1	1
233	0	0	0	0	0	0	0	0	0	0
234	0	0	0	0	0	0	0	0	0	0
235	0	0	0	0	0	0	0	0	0	0
---	-	-	-	-	-	-	-	-	-	-

c) Logical array of Binarize Image

Image Size: [251 833]
 Num of Objects: 14
 Pixel Index List: {[117852x1 double] [53129x1 double] [12021x1 double]
 [153x1 double] [376x1 double] [418x1 double] [379x1 double] [416x1 double]
 [152x1 double] [424x1 double] [900x1 double] [418x1 double] [149x1 double]
 [158x1 double]}

d) Results of connected component analysis

labeled = 251x833 uint8 matrix

	1	2	3	4	5	6	7	8	9	10
151	1	1	1	1	1	1	1	1	1	1
152	0	0	0	0	0	0	0	0	0	0
153	0	0	0	0	0	0	0	0	0	0
154	0	0	0	0	0	0	0	0	0	0
155	0	0	0	0	0	0	0	0	0	0
156	2	2	2	2	2	2	2	2	2	2
157	2	2	2	2	2	2	2	2	2	2

e) labelling matrix of the connected component



f) RGB image of labelling matrix (Each object appears in a different colour)

Figure 5-2. Steps of Connected Component analysis

```

I = imread('Revision.png');
figure
imshow(I)
title('Original Version of Image')
B = rgb2gray(I);
figure
imshow(B)
title('grayscale Version of Image')
BW = imbinarize(B);
figure
imshow(BW)
title('Binarize Version of Image')
z = logical(BW)
cc = bwconncomp(BW,26)
labeled = labelmatrix(cc)
RGB_label = label2rgb(labeled,@parula,'C','shuffle');
imshow(RGB_label)

```

Figure 5-3. Sample code of connected component in Matlab

After applying connected components, the model needs to detect text regions. There are several methods for text regions detection, such as maximally stable extremal region (MSER) and region of interest (ROI). Detect text regions using the MSER algorithm and remove non-text areas. The MSER algorithm was designed for blob detection in images[98]. Also, it is used for text detection to determine text regions from the image [99]. Also, ROI can be provided by defining one or more rectangular regions of interest around the text. After text regions detection, bounding boxes around words will be expanded and dilate images to make letters thicker. In the next step, the OCR function is applied to find as much text as possible in no specific order, even if embedded in images. The OCR functions will return the recognized text and its confidence level. Matlab OCR engine can provide the word confidence and character confidence for the extracted result. To have better accuracy, the model only accepts an 80% confidence level or more. Finally, the detected text will be stored in text file format

Figure 5-4 illustrates the different steps of text extraction, and Figure 5-5 illustrates the example of text extraction.

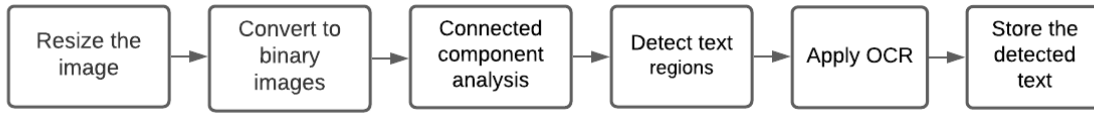


Figure 5-4. The process of text extraction

TITLE						
Isometric Drawing						
CLIENT DRAWING No. 0.5-FR-PSC2XR-164-0006-PK- 1						REV. 00A
PROJECT No.	AREA	DISC	DOC	DRAWING No.	SHEET	
334562	0162	46	D9	FR0006	1 of 1	
PROJECT VAULT				PROJECT No. 334562		
PROCESS PIPING				ENGINEER		
AREA 102 - CRYSTALLIZATION & REAGENT MIXING				ENGINEER/DESIGNER		
FROTHER STORAGE & DISTRIBUTION				REVISOR		

a) Resize

TITLE						
Isometric Drawing						
CLIENT DRAWING No. 0.5-FR-PSC2XR-164-0006-PK- 1						REV. 00A
PROJECT No.	AREA	DISC	DOC	DRAWING No.	SHEET	
334562	0162	46	D9	FR0006	1 of 1	
PROJECT VAULT				PROJECT No. 334562		
PROCESS PIPING				ENGINEER		
AREA 102 - CRYSTALLIZATION & REAGENT MIXING				ENGINEER/DESIGNER		
FROTHER STORAGE & DISTRIBUTION				REVISOR		

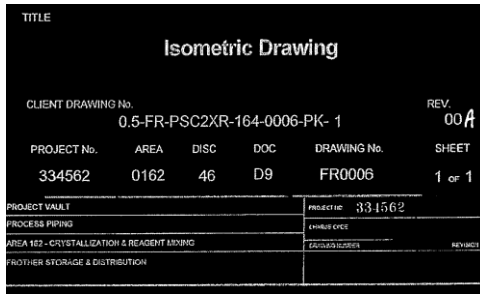
d) ROI

TITLE						
Isometric Drawing						
CLIENT DRAWING No. 0.5-FR-PSC2XR-164-0006-PK- 1						REV. 00A
PROJECT No.	AREA	DISC	DOC	DRAWING No.	SHEET	
334562	0162	46	D9	FR0006	1 of 1	
PROJECT VAULT				PROJECT No. 334562		
PROCESS PIPING				ENGINEER		
AREA 102 - CRYSTALLIZATION & REAGENT MIXING				ENGINEER/DESIGNER		
FROTHER STORAGE & DISTRIBUTION				REVISOR		

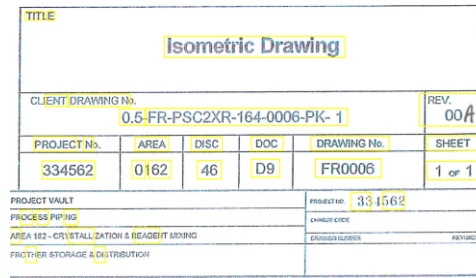
b) Binarize

TITLE						
Isometric Drawing						
CLIENT DRAWING No. 0.5-FR-PSC2XR-164-0006-PK- 1						REV. 00A
PROJECT No.	AREA	DISC	DOC	DRAWING No.	SHEET	
334562	0162	46	D9	FR0006	1 of 1	
PROJECT VAULT				PROJECT No. 334562		
PROCESS PIPING				ENGINEER		
AREA 102 - CRYSTALLIZATION & REAGENT MIXING				ENGINEER/DESIGNER		
FROTHER STORAGE & DISTRIBUTION				REVISOR		

e) Expanded bounding boxes



c) Connected Component



f) Detected text

Figure 5-5. Example of text extraction steps

5.1.2 Classification based on TF-IDF and different algorithms

Several approaches for automation of document classification have been studied [5–11]. Based on the literature, there are two major approaches. The first approach uses structured CAD drawings as input, as the CAD drawings can be fed to the model as either 2D or 3D objects. The second approach uses documents that are rich in text, unlike engineering drawings. In addition, there is a growing need for document classification based on scanned documents, which consist of the drawings, specifications, layouts, etc., with different formats, qualities, and sizes. Automation of scanned document classification has not been covered in the literature, and the main focus of the literature has been on documents that contain lots of text, while the scanned documents such as drawings have limited text, and sometimes, two different drawing document types will be distinguished only based on one word. Therefore, in this phase, the performance of classification methods on documents with image formats that contain mainly different types of drawings with limited text contents was investigated. A dataset of 8000 construction documents is used to examine the performance of alternative classification methods. Pre-processing steps are applied to increase the image quality of the documents. This includes converting the images to grayscale first, adjusting their brightness and contrast, and cropping if necessary.

Later, optical character recognition (OCR) is used to create a text file for each document. One of the main challenges in this step is dealing with noisy OCR results. Layout analysis and region of interest techniques were applied to decrease the effect of noisy OCR results. In the next step, the term frequency-inverse document frequency (TF-IDF) technique is used [100]. Finally, several classification algorithms such as Linear SVC (Support Vector Classifier), Logistic Regression, Multinomial Naïve Bayes, and Random Forest are applied on TF-IDF results, and a comparison of their performance is conducted.

The objective of this study is to evaluate the accuracy of traditional document classification methods under the poor text content conditions of construction drawings where documents are available in an image format (scanned documents), and the document content is not text-rich (mainly drawings). The input to this study is a dataset of 8000 construction documents that represent different types of drawings for industrial construction projects. These documents are in image format (.png, or .jpg) and include different resolutions and noises.

Proposed Methodology:

This section describes the steps followed in this study to develop and test the classification models. These include (1) data collection and dataset preparation, (2) pre-processing steps for improving image quality; (3) cropping the images; (4) optical character recognition (OCR); (5) cleaning process and tokenization; (6) vectorize using TF-IDF; (7) application of supervised machine learning algorithms; and (8) test and evaluate different classifiers. Figure 5-6 illustrates the methodology of classification based on TF-IDF and different algorithms.

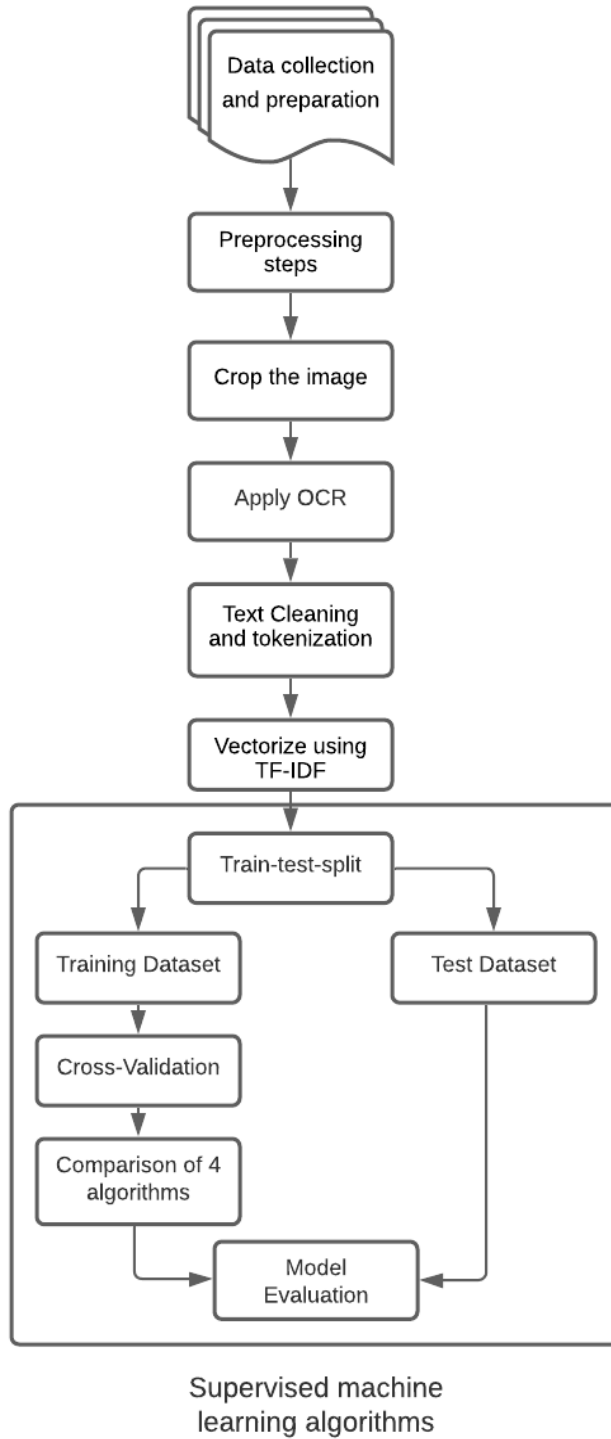
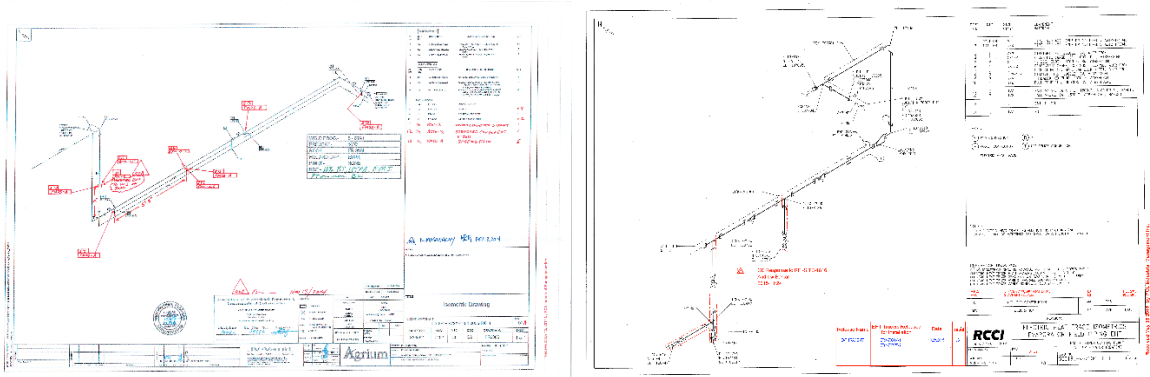


Figure 5-6. Methodology of classification based on TF-IDF and different algorithms

The current dataset used in this study includes eight classes of drawings common to industrial construction projects. These are electrical heat tracing (EHT), isometric, layout diagram, loop diagram, piping, and instrumentation diagram (P & ID), pipe support, single line diagram, and wiring diagram.

The dataset selection process started with selecting the images that had a dot-per-inch (dpi) value of 300 or higher to have a better data extraction using the OCR. Each of the eight classes is represented by 1000 documents in the dataset, making a total of 8000 documents. These documents were randomly selected by document management personnel in an industrial construction contractor firm from different construction projects completed by this contractor. All the documents are in image format (.png, or .jpg). Figure 5-7 shows sample documents in the dataset. As seen in Figure 5-7, a scanned document usually has a poorer resolution compared to a PDF document generated from a CAD authoring tool, and it may include background noise and textual noise, which has a negative effect on text extraction and classification process.



a) Isometric drawing

b) Electrical heat tracing drawing

Figure 5-7. Dataset samples for classification based on TF-IDF and different algorithms

Pre-processing steps for improving image quality

The case study includes images of the construction document. In the following steps, OCR was applied to images to extract all the text available in the document. Since image noise and low resolution can have a negative impact on OCR results, the quality of images needs to be improved before the OCR step. In this step, all the images are converted to grayscale images; and their brightness and contrast are increased. In addition, the resolution of images should be at least 300 DPI for a better text detection result.

Cropping the images

The information table contains the most important data about each engineering drawing, such as drawing name, project name, and numbers. Cropping the images and keeping the part that contains the table of information helps the model to have a faster performance as the number of pixels in each image is reduced. Therefore, there is less noise to be passed to the classifier. In engineering drawings, the table of information location varies depending on whether the image is horizontal or vertical. Usually, in a horizontal image, it is located on the right half, and in a vertical image, it is located on the bottom half of the image. A script was developed to process the document and perform cropping. First, the width and height of each image are extracted, and based on these variables, if the image is horizontal, the image gets cropped along its width, and if the image is vertical, it gets cropped along the height. Figure 5-8 shows the cropping of vertical and horizontal engineering drawings.

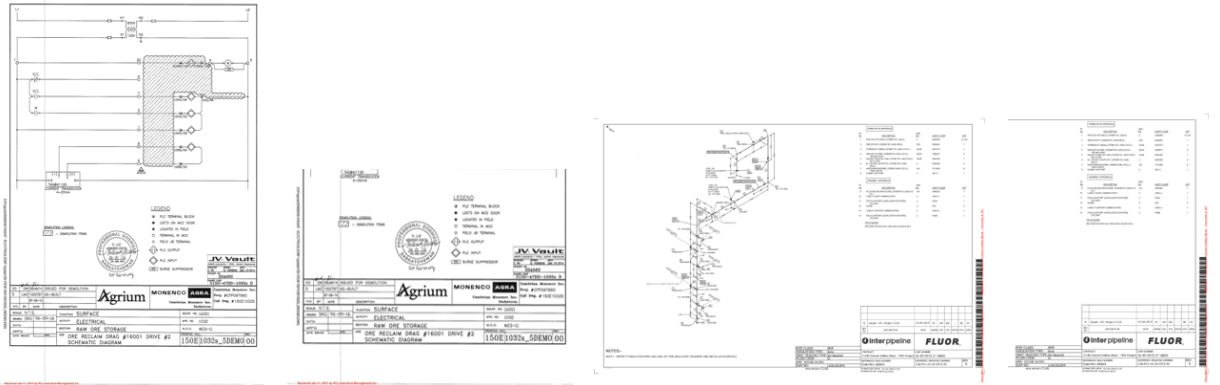


Figure 5-8. Example of cropping the images

Text extraction using OCR

OCR is the algorithm that recognizes text in images, and it is widely used for text extraction from documents and images [101]. In this research, the Tesseract OCR library in Python, one of the most accurate OCR engines, converts the images to a string.

Text Cleaning and tokenization

Due to the noise and fragmented text in the documents, many words are either unimportant or not recognized with reasonable accuracy by OCR. The cleaning process removes all the numbers, blank lines, special characters, single letters, and non-English words. In addition, all the extracted words were transferred to lower case to have consistent data, and they were filtered by their character length.

The text tokenization method is used to tokenize the text file into individual words in the next step. Tokenize is an operator that splits the text of documents into individual words called “word vector” based on token boundaries. Token boundaries are space and punctuation that can be defined differently based on the nature of each language [102]. In the next step, a dictionary of the NLTK library in Python was used to remove all non-English words[103]. Also, all the tokens are not qualifying for vectors can remove from the text. Such as stop word, which is not measured as keywords. It also will help to reduce the dimensionality of term space [104]. Also, the stemming process was applied, which means the words with different endings will be mapped into a single word, such as worker, workers, worked.

After cleaning the dataset, the code goes through each file in the directory, converts the images into a string, and saves them into a .csv file containing all the text for all the images, with their relative labels. Figure 5-9 illustrates the sample results of cropping the image, text extraction, and text cleaning and tokenization.

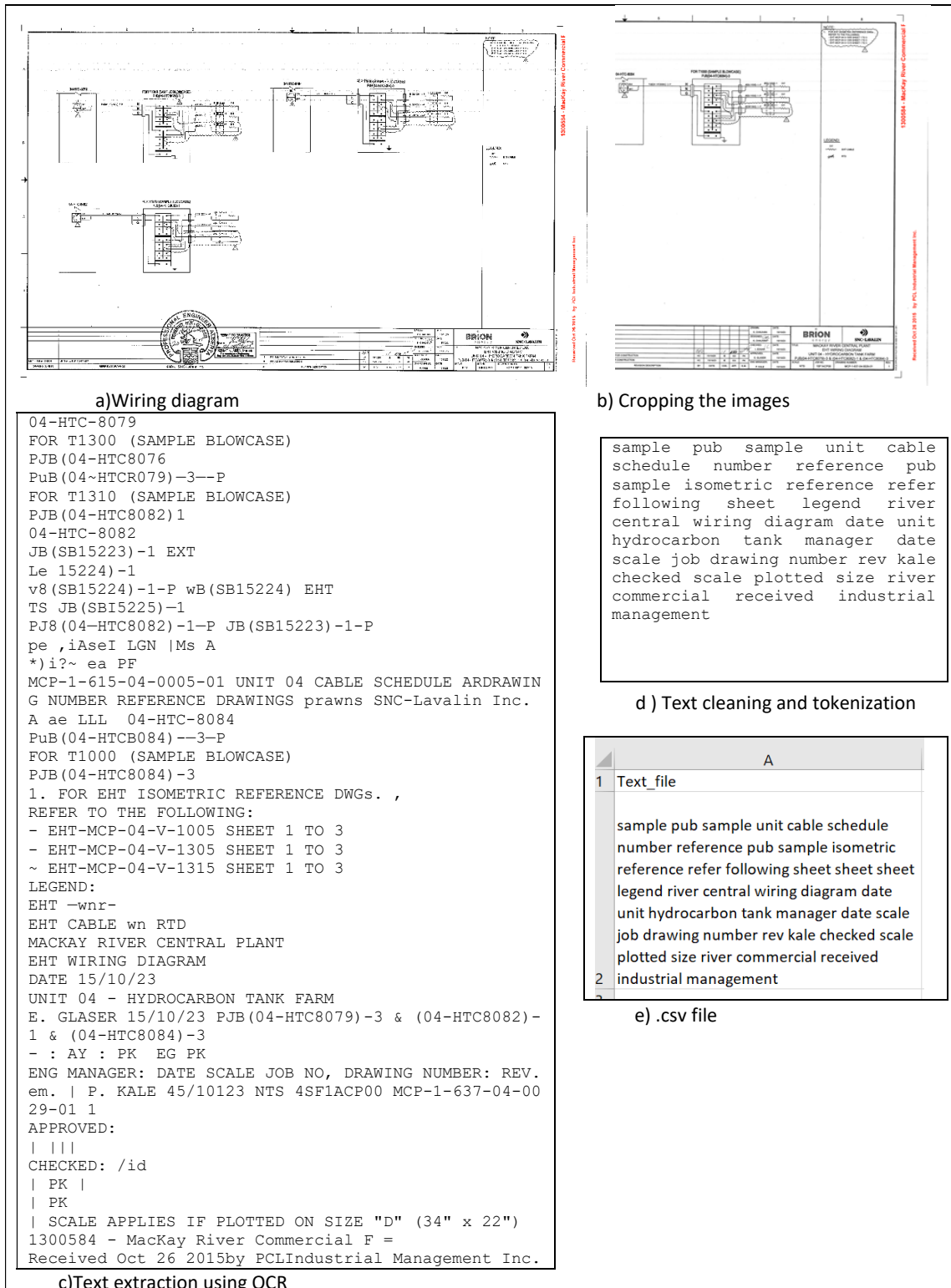


Figure 5-9. Sample results of cropping the image, text extraction, text cleaning, and tokenization.

Vectorize using TF-IDF

To reduce document dimension and convert the document into structure format TF-IDF (term frequency–inverse document frequency) is used. TF-IDF is one of the widely used weighting methods [105]. TF-IDF is a numerical statistic that reveals how much a word is important to a document in a collection. Frequency shows the number of words repeated in all documents, and document occurrences show the number of individual documents in which the word appeared [106]. TF-IDF factor equation of word t occurs in document d is [107]:

$$\text{TF-IDF}(d, t) = \text{TF}(t) * \text{IDF}(d, t) \quad (5-1)$$

Table 5-1 shows a TF-IDF sample for one document. The Term Index column values indicate a value for each term. In TF-IDF, each term is known with its respective index. To achieve the highest TF-IDF values, The TF-IDF scores are sorted from the highest to the lowest; more than 7000 terms were generated through TF-IDF, and 1000 of the highest TF-IDF scores were selected.

Table 5-1. TF-IDF samples for one document

Term Index	Term	TF_IDF score
0	train	0.43609242
1	coil	0.295912489
2	fort	0.285722496
3	fire	0.284439729
4	alarm	0.249922019
5	energy	0.247646536
6	interconnection	0.205736702
7	riser	0.192928212
8	normally	0.18905721
9	contact	0.170355013
10	provided	0.156025633
11	sea	0.156025633
12	circuit	0.133700617
13	ria	0.128437591
14	canada	0.120892127
15	client	0.110192193
16	corporation	0.104867052
17	mar	0.104131862
18	layout	0.103320581
19	permission	0.100927876
20	professional	0.097886535

Supervised machine learning algorithms

The last step is to classify the documents into their categories. Since each document's category labels are known, supervised machine learning algorithms were used in this research. From 8000 records available in the dataset, 70% (5600 records) are used for training, and the other 30% (2400 records) are used for testing. The splitting is performed randomly for all the classes; therefore, there are 700 selected records for training and 300 records for testing of each class. K-fold cross-validation is applied to the training data set for better performance to avoid overfitting during the training process. K-fold cross-validation is dividing the training set into k smaller sets. The model uses the k-1 fold for training and the remaining fold for validation purposes. This process will be repeated k times. Figure 5-10 illustrates K-fold cross-validation for k= 5.

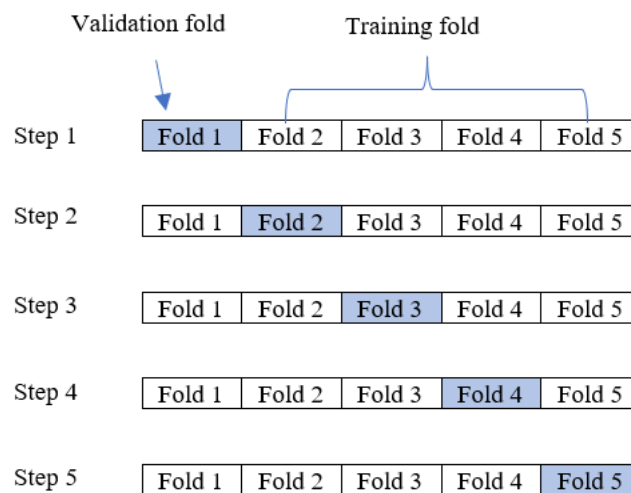


Figure 5-10. K-fold cross-validation

The training dataset contains a text file and label of each text file. Therefore, the model can learn from known data to predict the label for testing the dataset. Figure 5-11 shows a sample of the final dataset that is ready for training the model in the CSV file. K-fold cross-validation was applied by the Scikit-learn library in Python.

	A	B
1	Text_file	Label
2	instrument prefixed fire toxic gas annunciatic	Layout
3	face north instrument prefixed red fire toxic	Layout
4	face north face north goety degasser area fly	Layout
5	instrument prefixed fire toxic gas annunciatic	Layout
6	instrument prefixed fire toxic gas annunciatic	Layout
7	instrument prefixed fire toxic gas annunciatic	Layout
8	west pro wing limit instrument prefixed fire t	Layout
9	electrical shall comply electrical code part en	Layout
10	instrument prefixed fire toxic gas annunciatic	Layout
11	steam steam gen steam gen steam steam ste	Layout
12	instrument prefixed fire toxic gas annunciatic	Layout
13	red beacon fire drawing limit fire wang mana	Layout
14	red fire beacon fire drawing limit air air air in	Layout

Figure 5-11. Sample of the training dataset in CSV file

To compare the results of the classification, four different classifiers have been tested. The classifiers are Random Forest classifier, Linear support vector classifier (Linear SVC), Multinomial Naïve Bayes, and logistic regression.

The Random Forest classifier is a simple classification algorithm with accurate results, which is used widely for text categorization by other researchers [108]. It creates different samples from original data, applying decision trees to each sample[109]. Each decision tree has a vote for the prediction model, and the model with the most vote will be selected as a final model.

Linear SVC is a linear classification model based on a support vector machine (SVM) [110]. The idea of Linear SVC is dividing the data into different classes by fitting the line or "hyperplane" among the samples, and it uses a kernel function to find the optimal separating hyperplane [69].

Multinomial Naïve Bayes is another classification algorithm commonly used for document classification because of its fast and accurate algorithm [111]. Naïve Bayes is assumed that each word is independent, and the multinomial distribution represents the number of each word, and it counts the frequency of the words

in the document. The prediction model is also based on the word frequency information and features[112].

The Multinomial Naïve Bayes is an excellent choice for large datasets and multiclass prediction.

The logistic regression classifier works well to describe data and to explain the relationship among the data.

Unlike Naïve Bayes uses the weighted combination of the TF-IDF feature and the transforms are using the sigmoid function [113]. Implementation of classification algorithms was applied by the Scikit-learn library in Python [114].

Experiments and results

This section compares the performance of classification algorithms on our dataset. Four different classification algorithms (Linear SVC, Logistic Regression, Multinomial Naïve Bayes, and Random Forest) are used for the model. To compare the performance of algorithms, accuracy, recall, and precision were calculated. While accuracy represents the overall performance of the classification model, precision and recall are computed for each class separately. Many studies utilise accuracy as one of the most common metrics to evaluate the generalization ability of classifiers[115]. This method is fairly simple to understand and can be applied both to binary and multiclass classification problems. The accuracy of machine learning classification is considered to be an important aspect of this study. Accuracy can be calculated as:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (5-2)$$

The number of correctly classified positive examples is divided by the number of predicted positive examples to calculate the precision. High precision indicates that an example labelled as positive is indeed positive.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (5-3)$$

High recall indicates the class is correctly recognized. Recall is showing how many percentages of actual positive are recognized truly. As a result, the high recall indicates the class is correctly labelled. Since

higher recall means that most documents are labelled correctly, it is more important than the precision metric in this research

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (5-4)$$

TP, TN, FP, FN indicate True Positive, True Negative, False Positive, and False Negative, respectively.

In addition to quantitative metrics such as accuracy, recall and precision, qualitative metrics can be considered for the evaluation of machine learning. To assess the quality of an explanation, qualitative metrics can be applied such as simplicity, flexibility for addressing a range of classification problems, stability, and interpretability. Interpretability can allow developers to understand how the machine learning algorithm is working, and it will help to get informed decisions about how to improve the algorithm [116].

Table 5-2 shows the accuracy of the classification algorithms. The results show that the Linear SVC has the best performance by achieving 97% accuracy. Logistic Regression and Multinomial Naïve Bayes have the same accuracy, around 96%. Random Forest Classifier showed 92% accuracy. Since the accuracy of Linear SVC, Logistic Regression, and Multinomial Naïve Bayes are very close, also their precision and recall results were compared. Table 5-3 and Table 5-4 show the class precision and recall for different document types. The results show that: (1) Linear SVC resulted in a higher precision of 98% to 100%, Multinomial Naïve Bayes reached a precision of 91% to 100%, logistic Regression achieved a precision of 96 to 100%, and Random Forest achieved a precision of 75% to 100%. Random Forest precision is the lowest compared to the other three algorithms. (2) Linear SVC also resulted in a high recall of 98% to 100%, Multinomial Naïve Bayes reached a recall of 92% to 100%, logistic regression achieved a recall of 97 to 100%, and Random Forest achieved a recall of 84% to 99%. Random Forest recall is also the lowest compared to the other three algorithms. (3) Overall, Linear SVC achieved the best performance in accuracy, precision, and recall compared to the other algorithms.

Although all four algorithms performed similarly, Linear SVC emerged as the winner since it can handle unstructured and semi-structured data, such as the text of construction documents. In addition, linear SVC

minimizes overfitting problems compared to other algorithms [117]. However, there is a possibility that each of these four algorithms will work better on other samples or other documents. Linear SVC was chosen in this study based on the sample set presented.

Table 5-2. Accuracy of the four classification algorithms

Classifier	Accuracy
Linear SVC	0.97
Logistic Regression	0.96
Multinomial Naïve Bayes	0.96
Random Forest Classifier	0.92

Table 5-3. Precision results of the four classification algorithms

Document Type	Random Forest	Linear SVC	Multinomial Naïve Bayes	Logistic Regression
EHT	100	100	100	100
Isometric	99	99	99	99
Layout	96	98	97	98
Loop Diagram	99	100	99	100
P&ID	75	98	91	96
Pipe support	97	100	99	98
Single Line Diagram	96	100	100	100
Wiring Diagram	82	98	92	98

Table 5-4. Recall results of the four classification algorithms

Document Type	Random Forest	Linear SVC	Multinomial	Logistic
			Naïve Bayes	Regression
EHT	99	100	100	100
Isometric	99	99	99	98
Layout	88	99	94	98
Loop Diagram	98	100	100	100
P&ID	90	98	96	98
Pipe support	88	99	98	98
Single Line Diagram	84	98	92	97
Wiring Diagram	84	100	96	100

Evaluation of TF-IDF approach

In this phase, a text classification method is applied to classify construction drawings documents that are available in image formats either because of scanning of hard copies or protected output of CAD authoring tools. The study used a dataset of 8000 randomly selected documents to represent eight different types of drawings common to industrial construction projects. The method applies OCR techniques for text extraction and TF-IDF for text representation. The classification of documents is performed using four different algorithms. A comparison between Linear SVC, Logistic Regression, Multinomial Naïve Bayes, and Random Forest shows that Linear SVC, Logistic Regression, and Multinomial Naïve Bayes all perform at a reasonable accuracy that ranges between 97% to 96%. However, the Linear SVC has the best results on class recall and precision.

5.1.3 Classification based on a pre-defined set of keywords

Dataset preparation

Ten different document types and 500 documents from each document type (in total, 5000 documents) were selected for case study analysis. All documents were in PDF format, which should be converted to image format for the purpose of using OCR text detection. Adobe Acrobat Pro DC software was used to convert PDFs to PNGs. Since the information needs to extract is usually on the first page of each document, only the first page of each document was selected for future text analysis. Figure 5-12 illustrates a sample of the dataset.

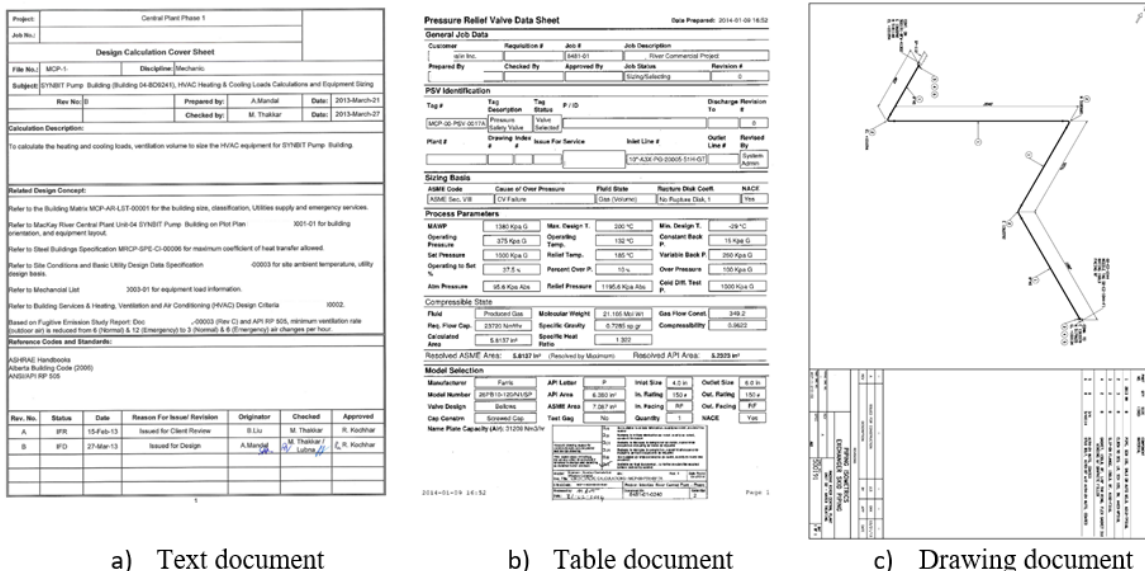


Figure 5-12. Dataset samples for Classification based on a pre-defined set of keywords

To process text categorization, a list of keywords was provided. The keywords can either be determined by experts in the specific domain of the application or based on historical documents. Table 5-5 illustrates the list of keywords that were used.

Table 5-5. keywords of ten classes

DOCUMENT TYPE	Description Keywords				
EHT zone drawing	EHT	Electric	Heat	Tracing	Iso
Isometric	Piping	Isometric	Iso		
Layout/location	Layout	Plan	Location		
Loop diagram	Loop	Diagram	Instrument		
Calculation	Calculation				
Datasheet	Data	Sheets	Instrument	Specification	
General arrangement	GA	General	Arrangement	Assembly	
Wiring diagram	Wiring	Diagram			
Schematic	Schematic	Power	Electrical		
Detail	Detail	Standard	Installation	Section	

In the experiment, a Matlab toolbox was used to find the keywords in the text files and then assign the appropriate classes to the documents. Based on the experiment, 5000 documents for text categorization from three different construction projects were analyzed. The classification accuracy is tested based on 100 documents for each class, and it was between 73% to 85% for different classes. However, in this method, each document can be assigned to more than one class since each document can contain the keywords of more than one class. The other challenge of this method is the reliability of accuracy. The accuracy of classification for a new project that we do not have any samples is questionable. To increase the reliability of the pre-defined keywords, text mining was applied via Matlab's Text Analytics Toolbox™. The purpose of this method is to analyze the most repeated words in each document type, and then unique words were added to the keywords, which can be another option instead of pre-defined keywords for any new project. In this step, the accuracy of classification has increased between 85% to 92%.

5.1.4 Classification based on deep learning–LSTM

A recurrent neural network (RNN) is a class of artificial neural networks which is developed during the 1980s [14]. RNN is a repeating neural network model with different loops to connect the previous information to the present task. When there is a gap between relevant information and output, the RNN cannot connect the information. A long short-term memory (LSTM) network is a type of RNN designed to fill the gap and avoid the long-term dependency problem. LSTM network will take three decisions about the information: decide about the useless information which should be removed, decide about the new information which should pass to the next layer, and decide about the output of each layer [77]. Recently, LSTM has been increasingly used to classify text data. Text data is naturally sequential, and LSTM can learn sequences from the training data [118]. Several researchers reported the high accuracy of their text classification result based on LSTM, such as [119–121]. Matlab’s Deep Learning Toolbox™ was used to test the LSTM network with a word embedding layer on our dataset. The dataset was imported to the toolbox in CSV format and contains two columns: The first one is the label of document types, and the second column contains the text of each document. In the next step, each word is converted to numeric sequences to be used as an input in the LSTM network. The LSTM model was defined by hidden layers and word embedding layers. Table 5-6 shows the architecture of the LSTM network used in our case study. The training model partitioning is 70% training, 15% validations, and 15% test observations. The maximum number of epochs is 30, and the initial learning rate is 0.01. The validation accuracy of classification is between 75% to 83%.

Table 5-6. LSTM architecture

Sequence input layer	1 dimension
Word Embedding Layer	50 dimensions and 6753 unique words
LSTM layer	80 hidden units
Fully connected layer	Number of classes, 10 layers
SoftMax layer	SoftMax
Classification Output layer	Number of outputs, 10 classes
Loss function	Cross entropy

Results and evaluation of LSTM approach

LSTM classification was repeated two times. The input of the first test was the original extracted OCR without cleaning process, and the input of the second test was cleaned OCR text. The accuracy of the first test was 80.93%, while the accuracy of the second test was increased to 94.38%. Figure 5-13 illustrates the training progress. The results show that LSTM has better accuracy on clean sets of data, and compared to pre-defined keywords, it does not require data analysis. The result of classification can improve if we can increase the accuracy of OCR text detection.

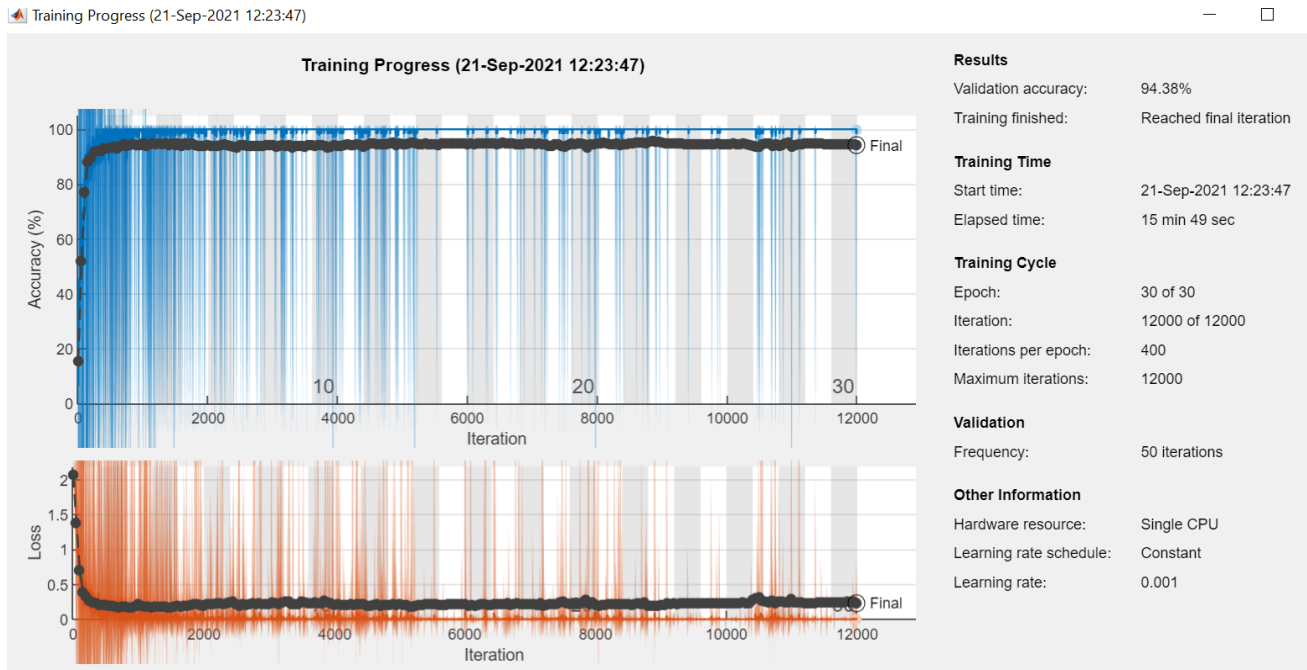


Figure 5-13. LSTM training progress

5.1.5 Comparison of results

The results of Stages 2, 3, and 4 illustrate that

- classification based on TF-IDF and Linear SVC achieved an accuracy of 97%;
- classification based on a pre-defined set of keywords achieved an accuracy of 85% to 92%; and
- classification based on deep learning–LSTM achieved an accuracy of 94.38%.

Classification based on TF-IDF and Linear SVC has the best results. The implementation outcome shows that despite the poor text content of drawing documents and the lack of structure of this content, the method used still performed very well with an accuracy of 97%, which indicates a potential solution for automating the classification task of such documents.

The result of classification based on LSTM shows that deep learning algorithms such as LSTM have the potential benefit of being used in construction documents classification. However, other deep

learning algorithms need to be tested. In this research, the experiment's result was considered as a preliminary result, and more data and algorithms should be tested in future research.

The pre-defined set of keywords is a rule-based approach that does not require training. However, the layout of documents and texts should be analyzed in advance, which is time-consuming and expensive. Also, each document can be assigned to more than one class since each document can contain the keywords of more than one class.

5.2 Phase 2: Developing classification based on image

Developing classification based on the image was designed based on three stages. Figure 5-14 shows three stages of phase 2. The first stage includes object detection API and classification algorithms. The second stage includes machine learning and a deep learning algorithm: AlexNet. The third stage is comparing the results of previous stages. The following section will discuss the stages in detail.

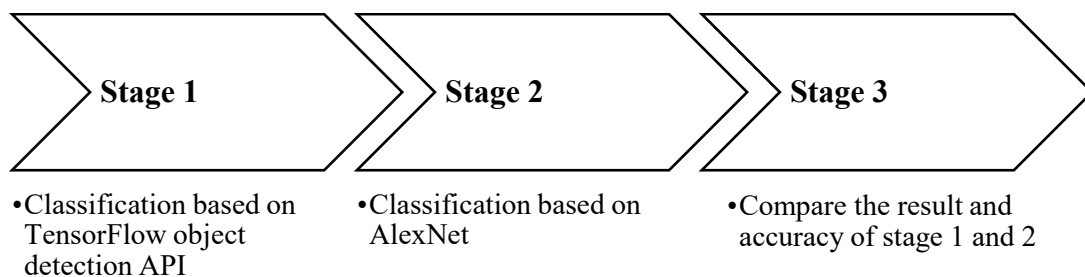


Figure 5-14. Methodology of construction document classification based on image

5.2.1 Classification based on TensorFlow object detection API

TensorFlow API experimental setup

The following TensorFlow API Setup was used for this research:

2x GeForce GTX 1080 Ti was used as a GPU model and 64 GB memory. For the graphical image annotation tool, LabelImg 1.80 was used. Anaconda Python 3.7 was used as the main Integrated Development Environment (IDE). TensorFlow 1.14 was chosen as an open-source machine learning framework. Linux Ubuntu 16.04 was used as an OS platform. The entire training of the CNN model was undertaken on Ubuntu. Detection model: faster_rcnn_inception_v2_coco. Also, Tensorboard was used for the visualization of the model.

TensorFlow API proposed methodology

The original document dataset exists in .pdf format, which should be converted to .png or .Jpg format.

Since documents have different sizes, three options can be considered:

1. Keep the original size, then label and change the size during the training process
2. Resize the images to 600 *1024, which recommended by Faster RCNN (resizer.py)
3. Crop the images and keep only the segment containing the title block, then resize (split.m)

After resizing the documents, the following steps and Python programs were undertaken [74]: First, all the documents are labelled. LabelImg was selected as an annotation tool that will convert each image into an XML file. In the next step, the training and testing dataset were generated by splitting the images into test and training folders 30% to 70%. Then all the train XML files are converted into a single train CSV file (xml_to_csv.py). Then, all the test XML files are converted into a single test CSV file (xml_to_csv.py). In the next step, train CSV file converted into TFRecord (generate_tfrecord.py) and test CSV file converted into TFRecord (generate_tfrecord.py). Then, the label map is created and configured by (pipeline.py). Also,

faster_rcnn_inception_v2 was selected as the object detection model. faster_rcnn_inception_v2_pets.config was used for configuring the environment. In the next step, the model trained by (train.py) and the model exported and generate the frozen inference graph. Finally, the COCO metric (eval.py) was used for the evaluation of the model.

Classification

Object detection API was used for classifying construction documents based on their image. Two different approaches were defined for image-based classification:

1. Classification based on document types
2. Classification based on drawing and non-drawing

Both of these approaches were tested by TensorFlow object detection API, and their accuracy was compared. Test 1 is designed for classification based on document types. Test 2 is about classification between drawing and non-drawing documents. Test 3 is also about classification between drawing and non-drawing documents with more data compared to the test 2. TensorFlow as an open-source machine learning framework and LabelImg as an image annotation tool are used in the following tests. The number of documents and number of classes was changed during the three tests.

Test 1: Classification based on document types:

Drawing documents are very similar to each other, and sometimes one word shows their difference. The purpose of the first test is to examine if the model can classify the drawing documents based on their types. Four classes of drawings were selected for the first test: isometric drawings, layout drawings, schematic drawings, and wiring drawings. The sample dataset was taken as 4000 images with 1000 images of each class. Table 5-7 illustrates the dataset of Test1. The whole area of the document was labelled. LabelImg

1.80 was used for the graphical image annotation tool. Figure 5-15 shows the area of labelling, which is blue. Table 5-8 illustrates the architecture of Test 1.

Table 5-7. Data set of Test 1

4000 Drawing Document	1000 Isometric
	1000 Layout
	1000 schematic
	1000 Wiring

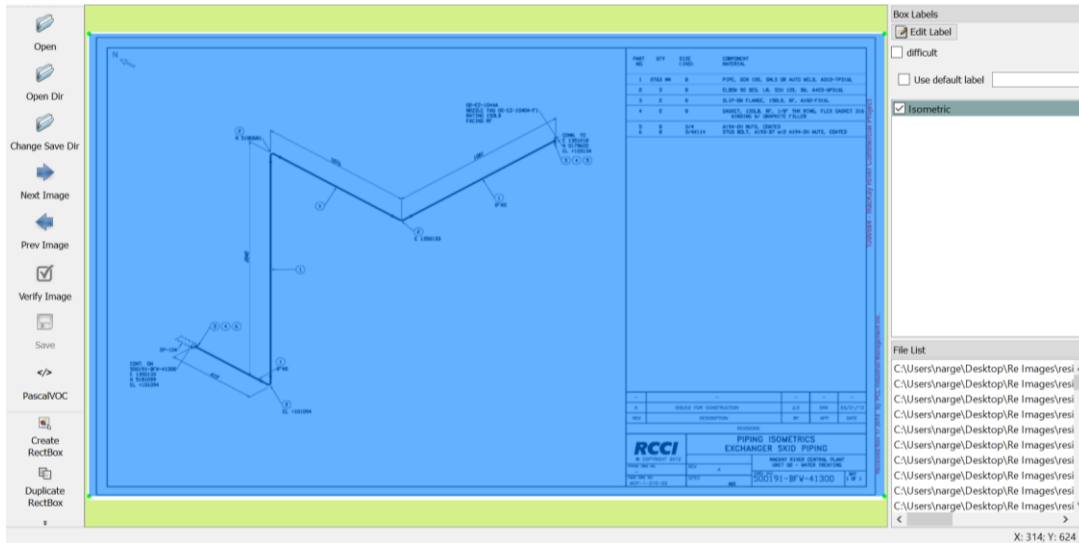


Figure 5-15. Sample of labelling

Table 5-8. The architecture of Test 1

Number of Documents	4000
Number of document type	4
Number of labels	1
Size of Documents	600 *1024
learning rate	0.0002
Number of steps	200000

The model has achieved 91% accuracy on isometric drawing, 92% on Schematic drawing, and it was failed to recognize document classes on Layout and Wiring documents because most of the images are very similar to each other's, and one or two words is their difference. Figure 5-16 and Figure 5-17 show the example output of the model. Figure 5-16 shows the result of the model on the isometric drawing is schematic and isometric. Figure 5-17 shows the result of the model on layout drawing is schematic and layout. The result of the first test shows that the model is not reliable for classification between drawing documents.

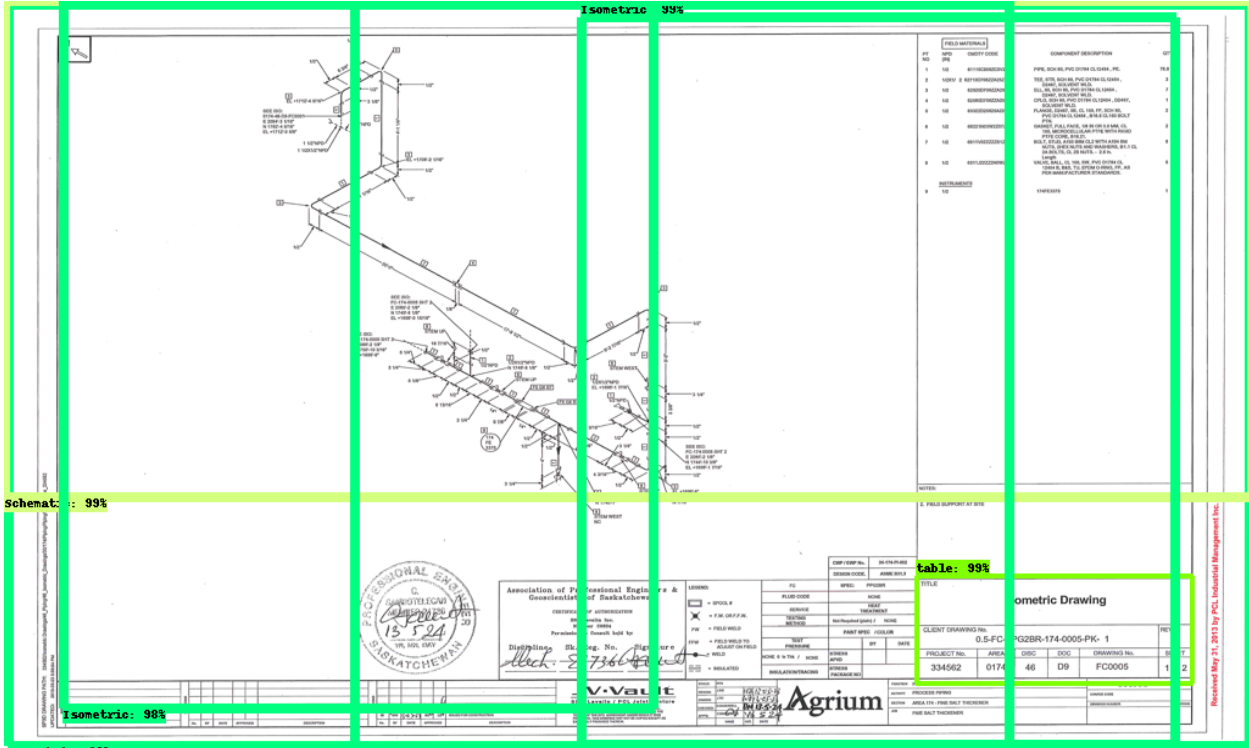


Figure 5-16. The result of the model on isometric drawing

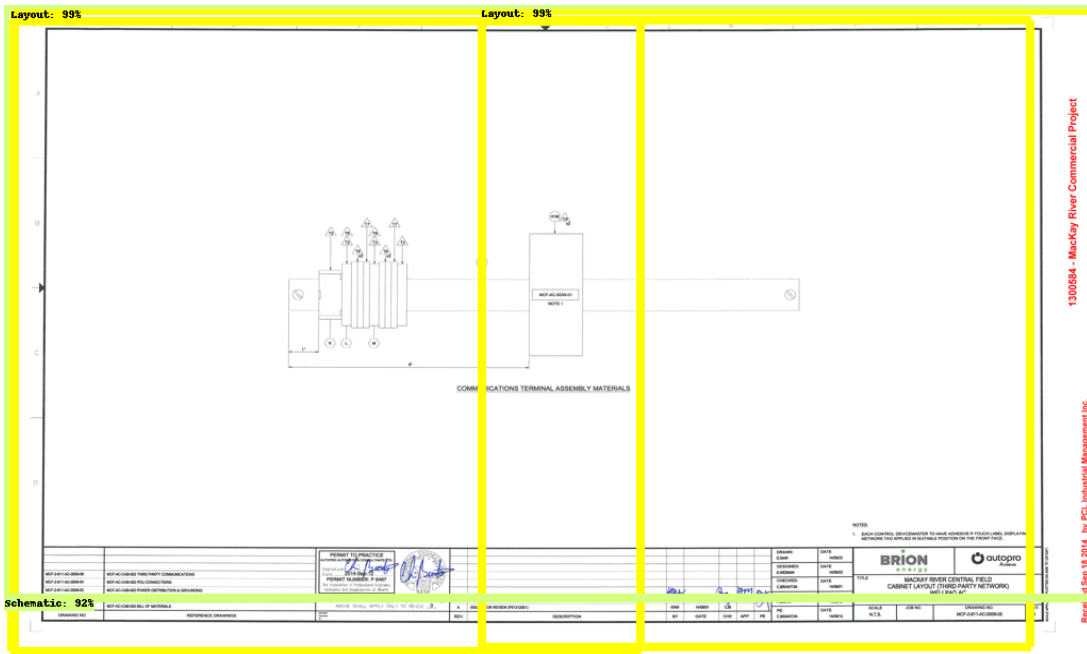


Figure 5-17. The result of the model on the layout drawing

Test 2: Classification between drawing and non-drawing documents

Usually, the difference between drawing documents and non-drawing documents is more evident. The drawing document includes an image and text and the table of information which is located on the right side. Non-drawing documents include tables and text. The purpose of the second test is to examine if the model can classify the documents into drawing and non-drawing types. For this test, 4000 documents were selected. 2000 images of drawing including isometric and schematic, and 2000 images of datasheet and work package as non-drawing were labelled.

Table 5-9. Data set of Test 2

4000 Document	2000 Drawing	1000 Isometric
		1000 Schematic
	2000 Non-drawing	1000 Datasheet
		1000 Work package

Table 5-10. The architecture of Test 2

Number of Documents	4000
Number of document type	2
Number of labels	1
Size of Documents	600 *1024
learning rate	0.0002
Number of steps	200000

The model is successful in recognizing document classes. Test evaluation shows average precision at 0.5

IoU: 0.84

EQUIPMENT DATA SHEET		DATA SHEET No.		Rev	DS
DISTRIBUTORS		00047241E0-08		Date	2016-03-16
Client: Agrum Project: Project VAULT		334662-0000-49E0-7241-08 <td>By</td> <td>JFL</td>		By	JFL
Project No. 334662		334662-0000-49E0-7241-08 <td>Appr</td> <td>JFL</td>		Appr	JFL
Project: Project VAULT		334662-0000-49E0-7241-08 <td>Appr</td> <td>CS</td>		Appr	CS
1	Equipment Name:	Cleaner Flotation Feed Distributor	Vendor:	Dentek Corporation	
2	Number Required:	1	Vendor Quotation Number:	Q-20032r4 SHC	
3	Equipment Number(s):	24734	Manufacturer:	Dentek Corporation	
4	Drawing Number:	100A7246	Model Number(s):	2-way 4" x 12" Flow Splitter	
5	Package Number:	PRC-46-7241			
6	Specification Number:	334662-0000-49E0-7241			
7	Process Description:	Two centrifugal pumps (one operating, one standby). Individual pipes from each pump feed slurry to the			
8	Slurry from the Cleaner Flotation Feed Pump Box is pumped to the distributor. The slurry is evenly distributed and gravity fed to parallel Cleaner	distributor banks. The equipment operates on a continuous basis with scheduled maintenance, 24 hours per			
9	day, 365 days per year.				
10					
11					
12	Total Flowrate to Individual Distributors:	Units	Value		
13	Feed - Nominal	m ³ /h	517		
14	Feed - Design	m ³ /h	509		
15	Feed - Nominal (Solids)	kg	64		
16	Feed - Design (Solids)	kg	63		
17	Slurry Velocity (Feed Pipe)	m/s	2.7 - 3.7		
18	% Solids	% (wt/w)	12.5		
19	Solids Specific Gravity		2.02		
20	Slurry Specific Gravity		1.24		
21	Slurry Viscosity		1.90		
22	Slurry Viscosity			Coarse Ore Slurry	
23	% Solids	% (wt/w)	62		
24	KCl	% (wt/w)	18		
25	FeCl ₃	% (wt/w)	0		
26	Inhibitors	% (wt/w)	0		
27	Aggregates	% (wt/w)	32		
28	FeCl ₃	% (wt/w)	200		
29	Inhibitors	% (wt/w)	0.01		
30	Particle Size Distribution				
31	Tyler Mesh	microns	% Passing		
32	4	8750	100		
33	6	3360	100		
34	8	2500	100		
35	10	2000	100		
36	14	1400	100		
37	20	850	100		
38	28	600	64		
39	35	420	48		
40	45	270	32		
41	60	140	14		
42	75	100	8		
43	100	74	5		
44	150	50	3		
45	200	37	2		
46	250	27	1		
47	300	20	0		
48	350	15	0		
49	400	11	0		
50	450	8	0		
51	Mixture Apparent Viscosity	Pa.s	1		
52	Number of Trays Required		1		
53	Number of Cycles Required		1		
54	Connecting Material (at Inlet)		316L SS		
55	Connecting Material (at Outlet)		316L SS		
56	Connecting Process Piping Material (at Outlet)		316L SS		
57					

non-engineering: 98%

non-engineering: 99%

non-engineering: 99%

Received May 08, 2015 by PCL Industrial Management Inc.

Figure 5-18. The result of the model on the datasheet

Figure 5-18 shows that the result of the model on the datasheet drawing is non-drawing which is correct.

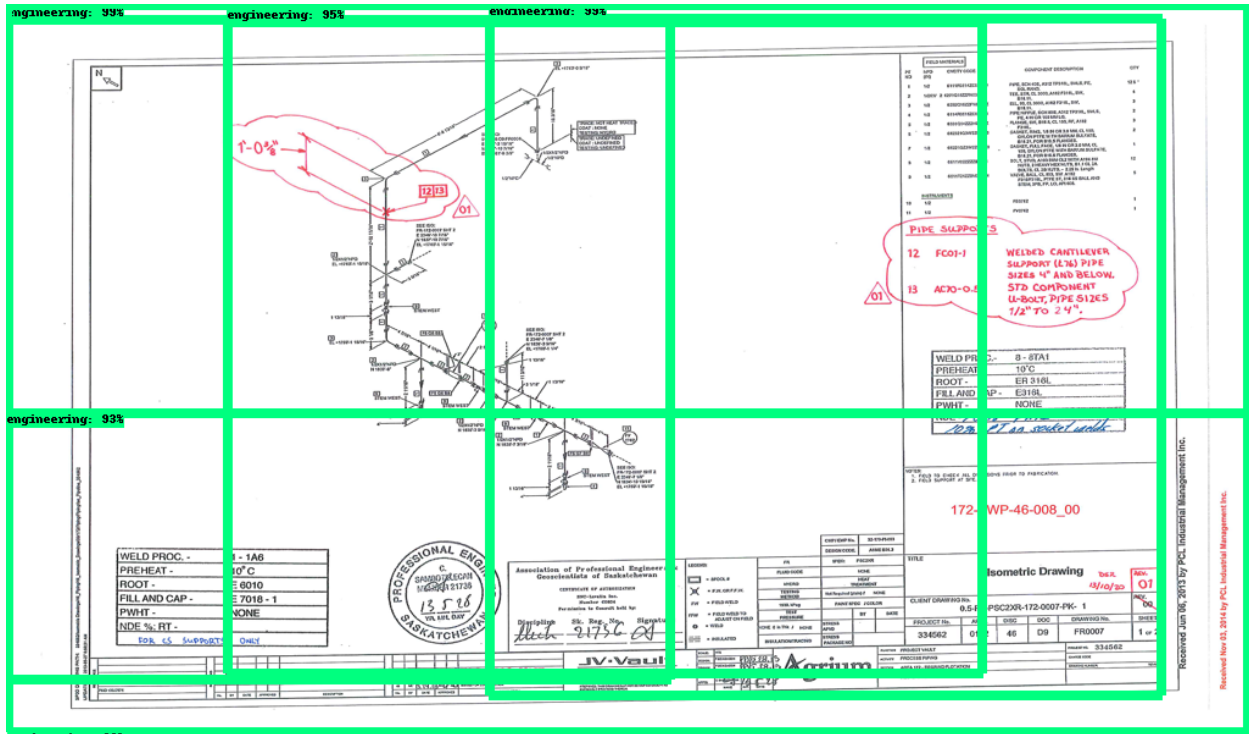


Figure 5-19. The result of the model on isometric drawing

Figure 5-19 indicates that the result of the model on Isometric drawing is drawing, which is correct.

non-engineering 375

JV Vault <small>SNC-Lavalin / PCL Joint Venture</small>	Engineering Work Package Driven Pile Testing	Revision		Page
		#	Date	
		01	2011-08-29	1
Document No.	000C7002EW	334562-0000-42EW-7002		
EWP No.	00-000-CI-001			

CLIENT: AGRUUM

PROJECT: VAULT PROGRAMME

		SIGNATURE	DATE
PREPARED BY:	A. Shehata	<i>A. Shehata</i>	2011/08/29
REVIEWED BY:	L. Leung	<i>L. Leung</i>	2011/08/29
APPROVED BY:	K. McLean	<i>K. McLean</i>	2011/09/30

non-engineering 385

ISSUE/REVISION INDEX

Issue Code	Revision					Revision Details
	No.	By	Rev'd.	App.	Date	
RC	00	AS	LL	KM	11-08-04	Released for Construction
RC	01	AS	LL	KM	11-08-29	Released for Construction

Issue Codes: RC = Released for Construction, RD = Released for Design, RF = Released for Fabrication, RI = Released for Information, RP = Released for Purchase, RQ = Released for Quotation, RR = Released for Review and Comments.

J:\314562\00-ENR343-STRN42-Concrete\EN-Scope of Work\334562-0000-42EW-7002 Rev 1 Driven Pile Testing_2011-08-29.doc EN-035C Rev 1 01

Figure 5-20. The result of the model on the work package

Figure 5-20 shows that the result of the model on work package drawing is non-drawing which is correct. The result of Test 2 illustrates that the model has better performance of drawing and non-drawing classification. Therefore, we need to test more data and include more document types.

Test 3: Classification between drawing and non-drawing documents

Test 3 examines if the model can classify the documents into drawing and non-drawing types with testing more data. Test 3 uses 8000 documents. 4000 images of drawing and 4000 images of non-drawing were labelled.

Table 5-11. Data set of Test 3

8000 Document	4000 Drawing	1000 Isometric
		1000 Schematic
		1000 Wiring
		1000 Layout
	4000 Non- drawing	1000 Datasheet
		1000 Work package
		1000 Bill of Material
		1000 Cable Schedule

Table 5-12. The architecture of Test 3

Number of Documents	8000
Number of document type	2
Number of labels	1
Size of Documents	600 *1024
learning rate	0.0002
Number of steps	200000

The model is successful in recognizing document classes. Test evaluation shows average precision at 0.5
IoU: 0.92

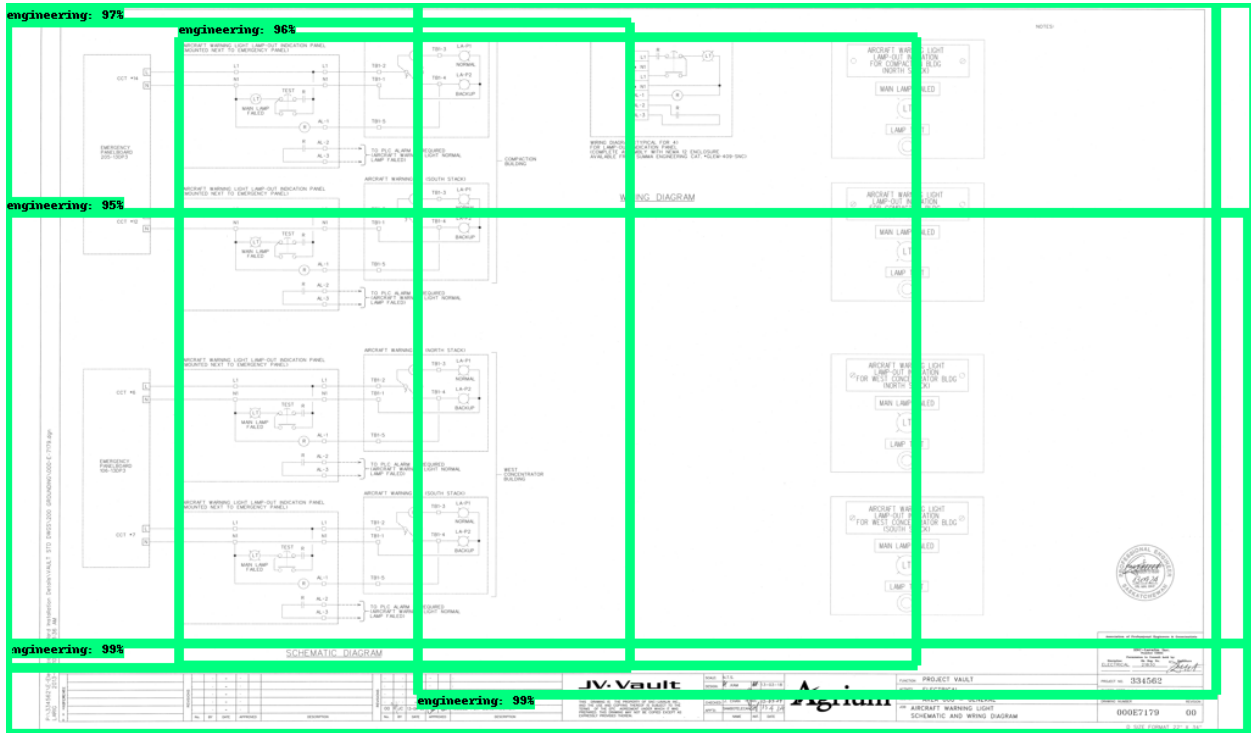


Figure 5-21. The result of the model on the schematic

Figure 5-21 shows that the result of the model on the schematic is drawing, which is correct.

non-engineering: 95%

BILL OF MATERIALS

Client: AGREUM		Project: PROJECT VAULT	
Reference Drwgs: 200A1269 South Dyke Remediation 200A1273 to 200A1276 Drainage Improvements		BoM Title: South Dyke Remediation and Stockpile Drainage Improvements 200C8001ET	
SNC-Lavalin / PCL Joint Venture		BoM No: 334562-0200-41ET-0001 Rev 00	
		Proj #: Chris Fortner Date: 2011-10-26	
		Checked: Date:	
		Apprvd:	

REV NO	ITEM NO	UNIT	QTY	DESCRIPTION	UNIT PRICE (\$)	TOTAL PRICE (\$)
South Dyke Slump Remediation						
0	1.1	m3	16,774	Surveyed Total Embankment Excavation		
0	1.2	m3	11,774	Imported Fill (Total Embankment Less Drainage Sand)		
0	1.3	m3	5,000	Actual Total Drainage Sand (Haul Truck)		
Brine Pond Stockpile Perimeter Drainage Improvements						
0	2.1	m3	25,100	Estimated Fill placement (Estimated based on surveyed original ground to modelled design surface)		
0	2.2	m3	6,950	Excavation (Estimated based on surveyed original ground to modelled design surface)		
0	2.3	m	15.5	0.6 m D1AM CSP culvert for stockpile drainage ditch		
0	2.4	tonnes	20.0	Granular base course		
0	2.5	m2	28.0	Woven Geotextile		
0	2.6	tonnes	14.2	Riprap at toe of slope		
0	2.7	m2	80.0	HDPE Liner below culvert outlet		

non-engineering: 99%

PAGE 1 OF 1

Received Nov 03, 2011 by PCL Industrial Management Inc.

Figure 5-22. The result of the model on Bill of Materials

Figure 5-22 illustrates that the result of the model on the Bill of Material is non-drawing which is correct.

non-engineering: 99%




**MACKAY CONTROL COMPLEX
CABLE AND CONDUIT SCHEDULE
UNIT 90 - CONTROL COMPLEX
MCC-90-CAB-09 FIBRE OPTIC CABLE SCHEDULE - PATCH CORD CABLING**

PROJECT	MACKAY CONTROL COMPLEX
DOCUMENT TITLE	MCC-90-CAB-09 FIBRE OPTIC CABLE SCHEDULE - PATCH CORD CABLING
DOCUMENT NUMBER	MCC-1-615-90-0005-01
DOCUMENT REVISION	0
DOCUMENT STATUS	ISSUED FOR CONSTRUCTION
DOCUMENT TYPE	CABLE AND CONDUIT SCHEDULE
OWNER/AUTHOR	JASMEET GILL, AUTOPRO
ISSUED DATE	21-Nov-14
DISCLOSURE	




REV.	STATUS	DATE	ISSUED FOR	BY	MM	YZ
0	IFC	21-Nov-14	ISSUED FOR CONSTRUCTION	JG	MM	YZ
A	IFR	12-Nov-14	ISSUED FOR REVIEW	JG	MM	YZ

non-engineering: 99%

1300584 - Mackay River Commercial Project
Received Nov 05 2014 by PCL Industrial Management Inc.

Figure 5-23. The result of the model on the cable schedule

Figure 5-23 shows that the result of the model on the cable schedule is non-drawing which is correct. The result of Test 3 demonstrates that by increasing the amount of data, the accuracy of the model is increased from 84% to 92%. Object detection API was not successful for document type classification, but it was successful for drawing, non-drawing classification.

5.2.2 Classification based on AlexNet

AlexNet experimental setup and proposed methodology

AlexNet is a CNN algorithm introduced by Alex Krizhevsky (2012) [72]. The literature review shows that it has outstanding achievements in deep learning and computer vision [122–126]. AlexNet algorithm has eight layers, and it relies on the structure and layout of the documents to classify them.

AlexNet classification was applied via Matlab Deep Learning Toolbox™. Dataset loading and labelling are the first steps. After loading the images in a folder, the image datastore function will automatically label the images based on folder names. Then, the data was divided into 70% for training and 30% for validation. In the next step, a pre-trained AlexNet neural network will load which includes more than a million images from the ImageNet database. The first layer will reduce the size of images to 227 by 227 by 3. The number 3 shows the colour channels. Also, the last three layers of the pre-trained network, which are a fully connected layer, a SoftMax layer, and a classification output layer [102], need to be changed based on the new classification problem. In the next step, the network will train based on training options such as epoch, mini-batch size, and validation data. Finally, in the validation section, the accuracy will calculate based on the fraction of labels that the network predicts correctly.

Classification

Similar to TensorFlow object detection API, two different approaches were defined for image-based classification:

1. Classification based on document types
2. Classification based on drawing and non-drawing

Both approaches were tested by AlexNet Network, the number of documents and the number of classes were changed during the three tests.

Test 1: AlexNet classification between drawing and non-drawing documents

Similar to Test 1 in TensorFlow, Test 1 is examined if the model can classify the documents based on their type. For this test, 4000 documents were selected, including 2000 images of drawing and 2000 images of the non-drawing document.

Table 5-13. Data set of Test 1

4000 Document	2000 drawing	1000 Isometric
		1000 Schematic
	2000 non-drawing	1000 datasheet
		1000 work package

Table 5-14. The architecture of Test 1

Number of Documents	4000
Number of classes	4
Size of Documents	600 *1024
learning rate	0.0001
epoch	6
Accuracy	88.89%

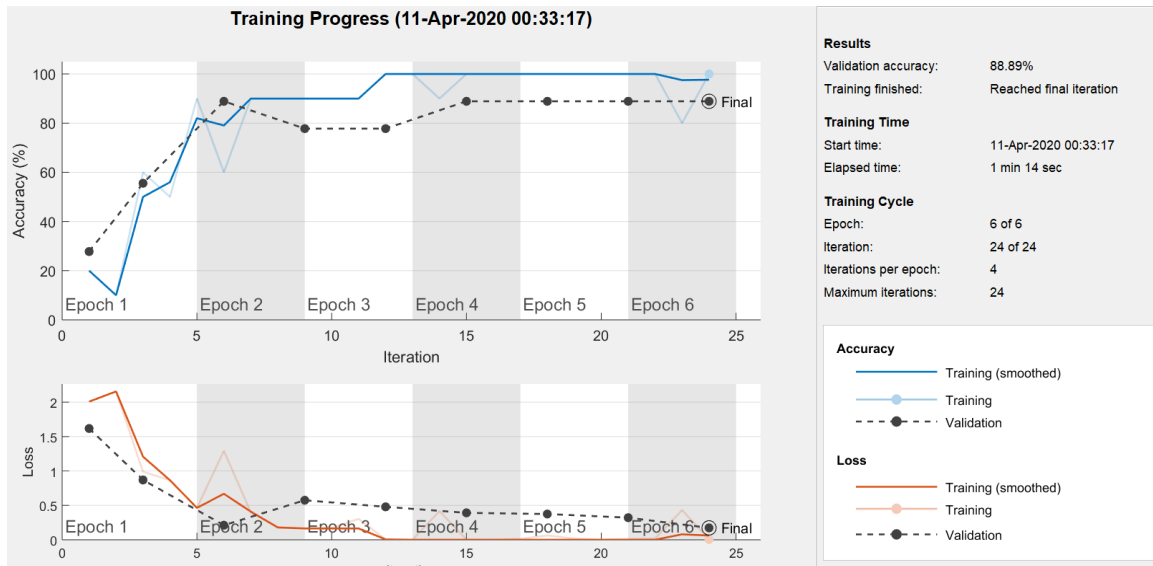


Figure 5-24. Training progress of Test 1

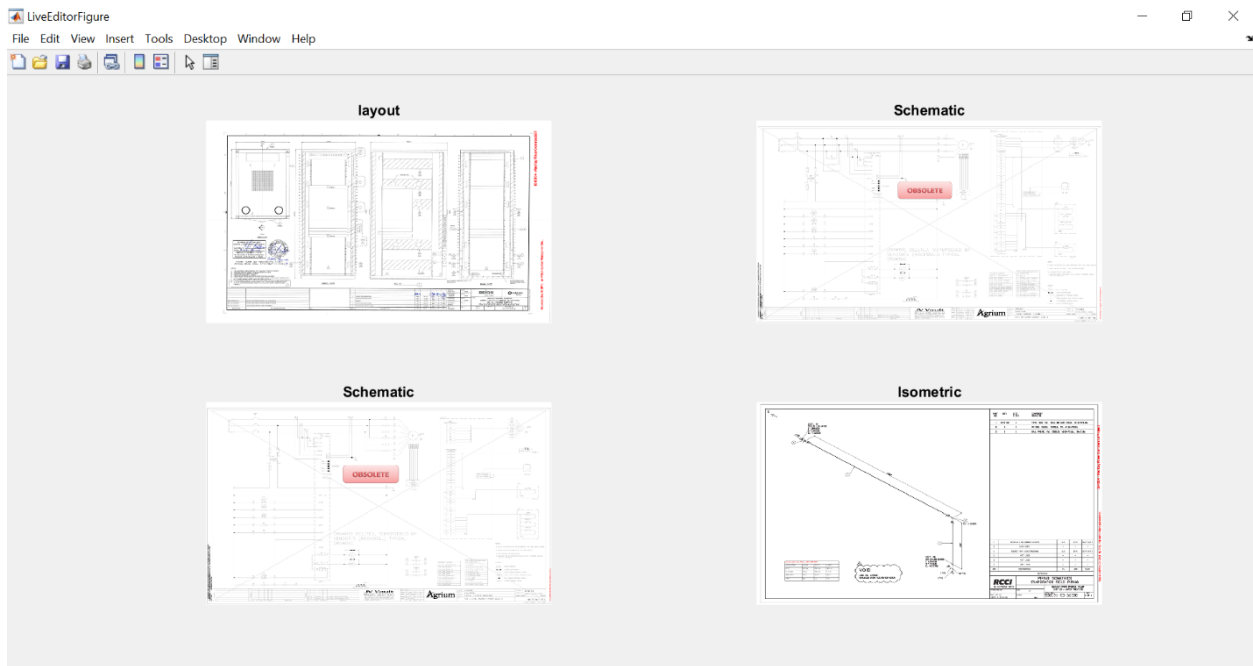


Figure 5-25. Result of Test 1

Test 1 was successfully classified documents based on their type. The accuracy of the model was 88.89%.

Figure 5-24 illustrates the training process of Test 1, and Figure 5-25 shows the result of Test 1, which

was able to classify layout, schematic, and isometric drawing correctly. Therefore, the amount of data in Test 2 is increased to see how the accuracy would change.

Test 2: AlexNet classification based on documents type

To test the accuracy of the classification model, the dataset is increased to 8000 documents. 4000 images of drawing and 4000 images of non-drawing were selected. Table 5-15 shows the dataset’s details.

Table 5-15. Data set of Test 2

8000 Document	4000 drawing	1000 Isometric
		1000 Schematic
		1000 Wiring
		1000 Layout
	4000 non- drawing	1000 Datasheet
		1000 Work package
		1000 Bill of Material
		1000 Cable Schedule

Table 5-16. Result of Test 2

Number of Documents	8000
Number of classes	8
Size of Documents	600 *1024
learning rate	0.0001
epoch	6
Accuracy	79.49%

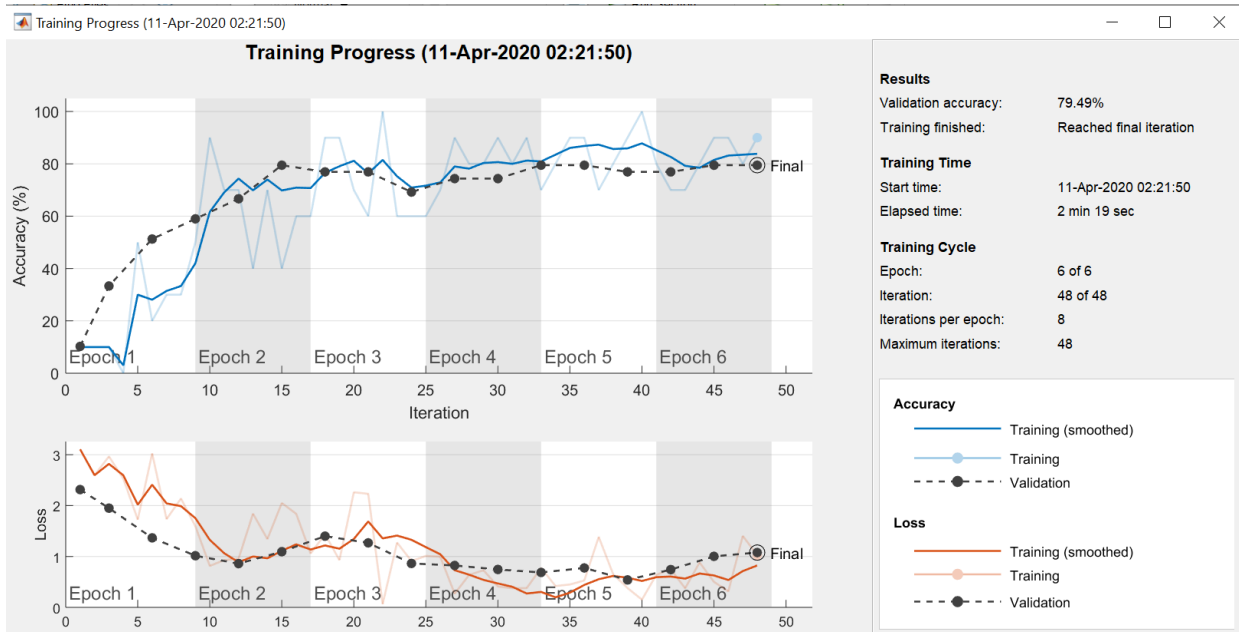


Figure 5-26. Training progress of Test 2

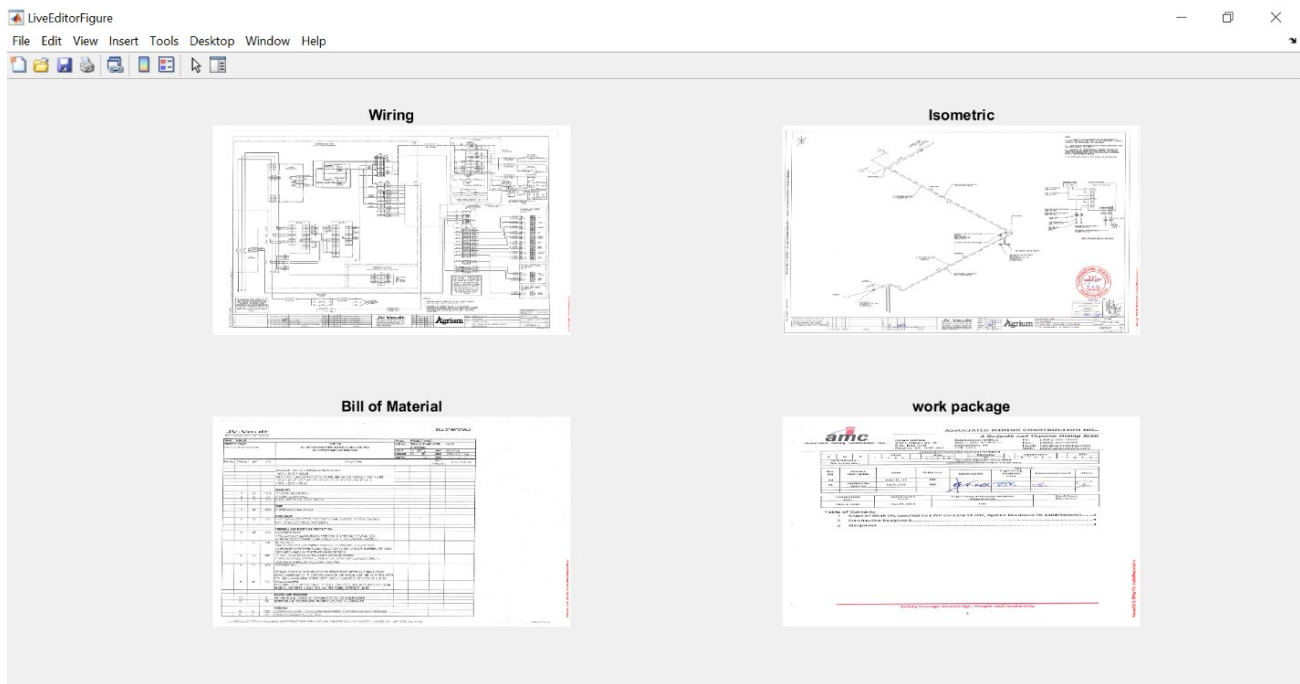


Figure 5-27. Result of Test 2

Test 2 was successfully classified documents based on their type with an accuracy of 79.49%. When the document types increased from 4 to 8, the accuracy of the model is decreased from 88.89% to 79.49%.

Figure 5-26 illustrates the training process of Test 2, and Figure 5-27 shows the result of Test 2, which was able to classify isometric drawing, bill of materials, work package, and wiring correctly.

Test 3: AlexNet classification between drawing and non-drawing documents

Test 3 is designed to test the classification accuracy between drawing and non-drawing documents.

Test 3 is the same as Test 2. Only the number of classes was changed to two classes. Therefore, the result will divide the engineering and non-engineering document. Table 5-17 illustrates details of the dataset.

Table 5-17. Data set of Test 3

8000 Document	4000 drawing	1000 Isometric
		1000 Schematic
		1000 Wiring
		1000 Layout
	4000 non- drawing	1000 datasheet
		1000 work package
		1000 Bill of Material
		1000 Cable Schedule

Table 5-18. Result of Test 3

Number of Documents	8000
Number of classes	2
Size of Documents	600 *1024
learning rate	0.0001
epoch	6
Accuracy	87.5%

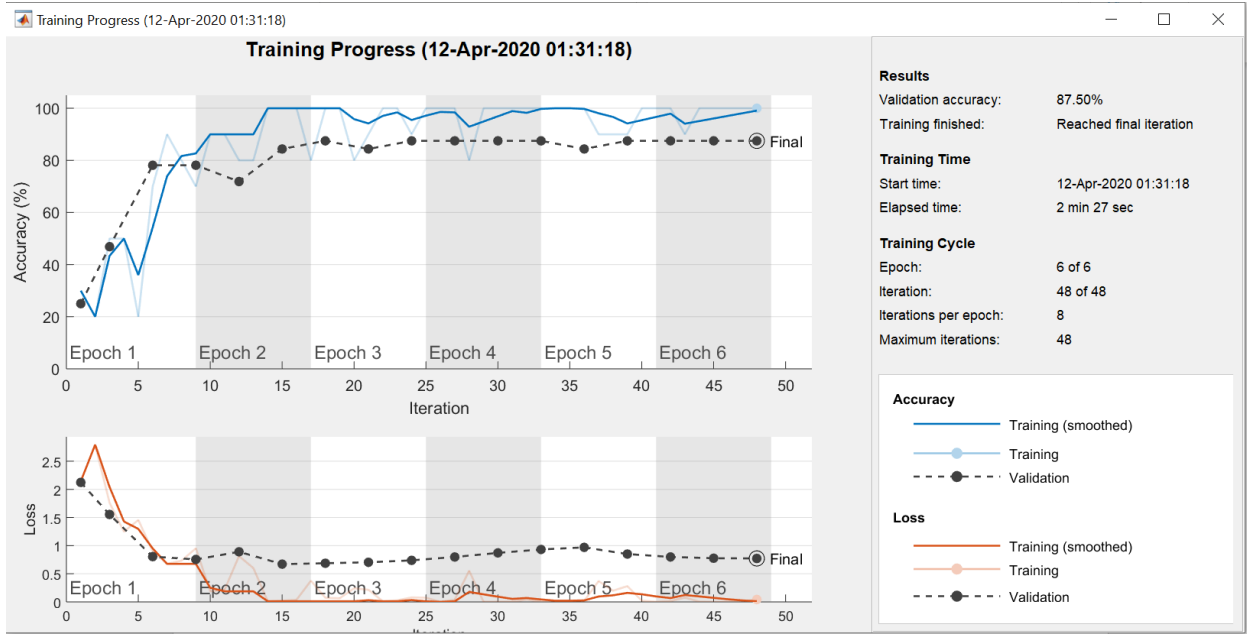


Figure 5-28. Training progress of Test 3

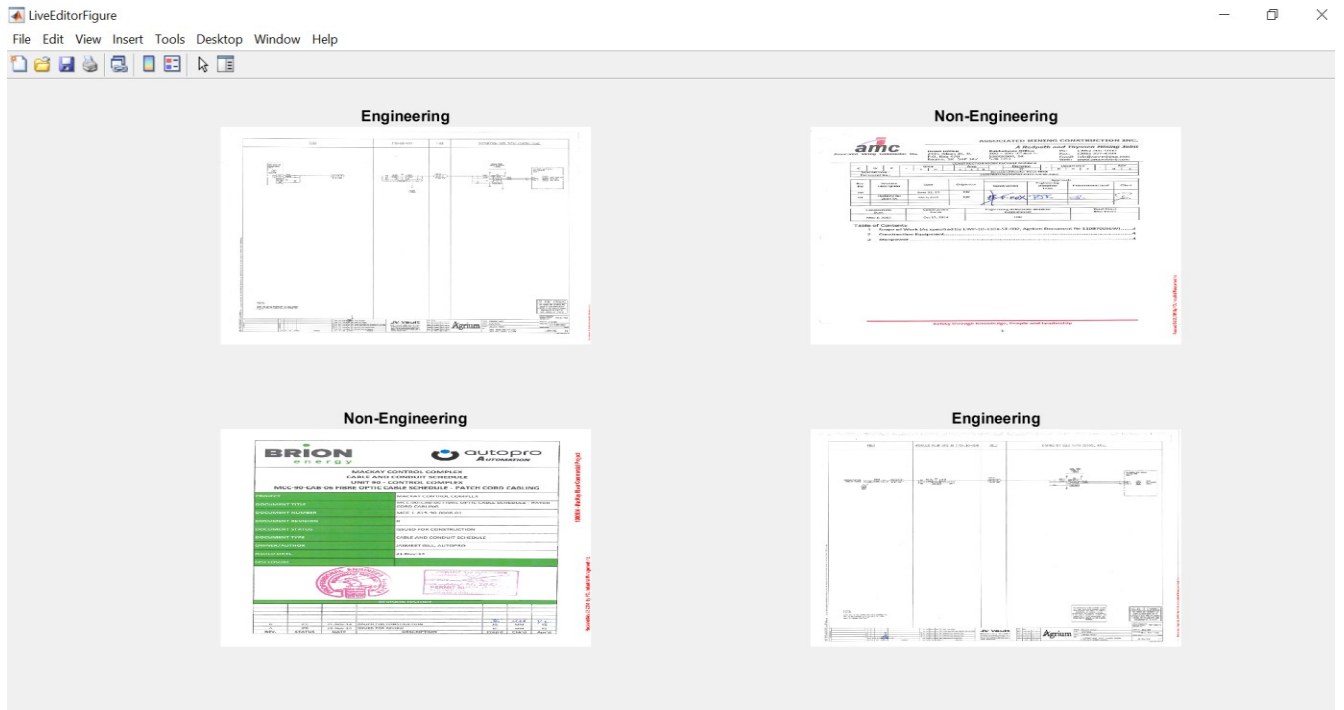


Figure 5-29. Result of Test 3-A)

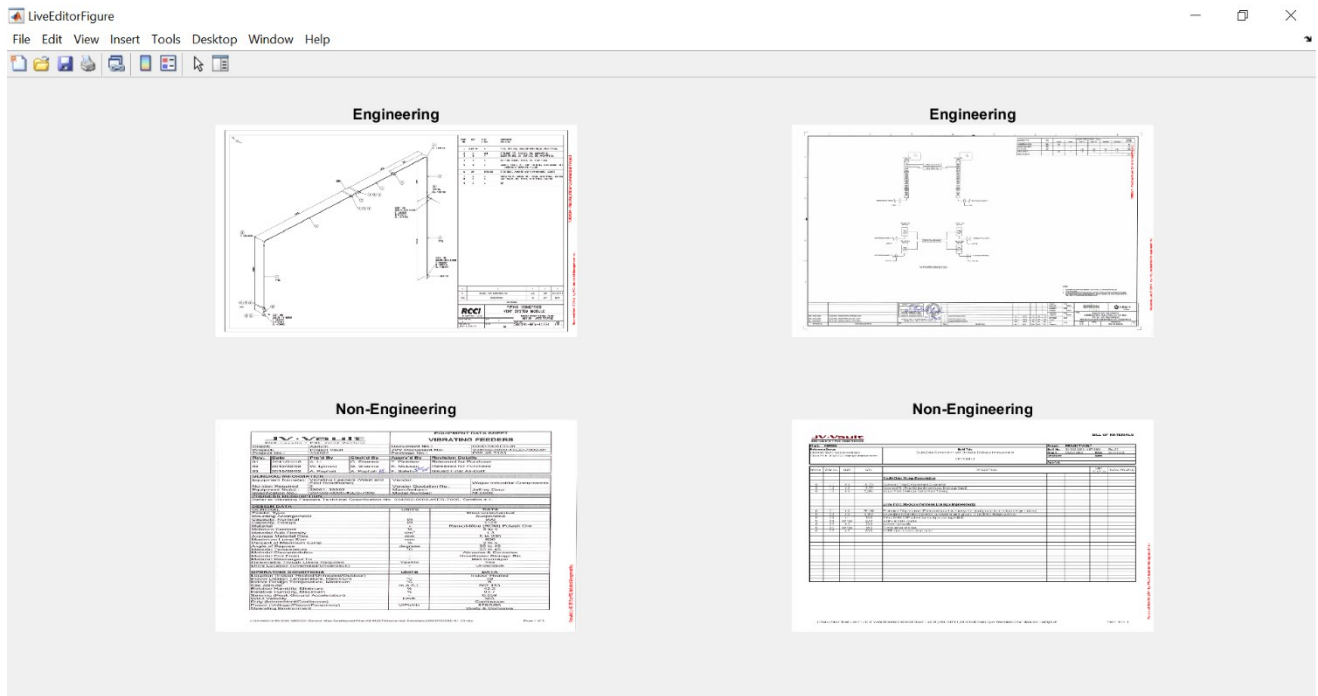


Figure 5-30. Result of Test 3-B)

Test 3 was successfully classified documents based on drawing and non-drawing with an accuracy of 87.5%. Figure 5-28 illustrates the training process of Test 3, and Figure 5-29, Figure 5-30 show the result of Test 3, which was able to classify drawing and non-drawing documents.

5.2.3 Comparison of results

Table 5-19 shows the summary of the result of TensorFlow and AlexNet tests. The result shows that TensorFlow object detection API is achieved the highest accuracy, which is 92%, for classification between drawing and non-drawing documents. Alex Net classification achieved better results on classification based on document types which TensorFlow object detection API was failed.

Table 5-19. Summary of the result of TensorFlow API and Alex Net

Test No	Classification based on:	Model	Number of Documents	Document Type	Accuracy
1	Document type	TensorFlow API	4000	4	Fail
2	Drawing and non- drawing	TensorFlow API	4000	2	84%
3	Drawing and non- drawing	TensorFlow API	8000	2	92%
1	Document type	Alex Net classification	4000	4	88.89%
2	Document type	Alex Net classification	8000	8	79.4%
3	Drawing and non- drawing	Alex Net classification	8000	2	87.5%

5.3 Phase 3: Developing title block detection and information extraction

Developing a title block detection model and information extraction were designed based on four stages. Figure 5-31 shows four stages of phase 3. The first stage includes the methodology of title block detection. The second stage includes experiments of object detection API to detect title blocks. The third stage includes the implementation and training of two tests. The fourth stage is evaluation and results. In the following sections, I will discuss each of the stages in detail.

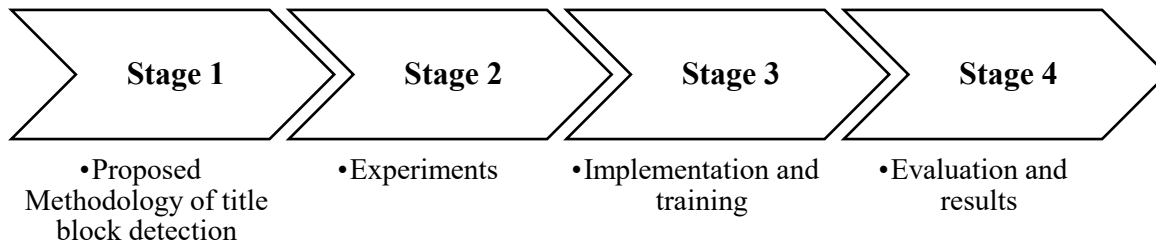


Figure 5-31. Different stages of developing title block detection and information extraction

Thousands of documents are generated through the life cycle of large construction projects. These documents need to be classified for document control purposes. Manual document control is time-consuming, and its accuracy can be subject to human errors. On large, fast-tracked projects, such tasks can consume the effort of a dedicated team and therefore, an automated solution can provide person-hour savings, support consistency, and increase the accuracy of document control processes. There are several methods and techniques available for construction document classification. Previous works in this field use the entire document text for classification and information extraction, which does not have high accuracy on all types of construction documents. To address these issues, I propose an approach to classification and information extraction based only on the title block, which has the essential information needed.

This chapter aims to evaluate the effectiveness of an automated method for title block detection and whether such an approach can be used to facilitate automated document classification and information extraction. The input to this automated approach is a set of scanned construction documents, including drawing and

non-drawing documents. On large construction projects, engineering services are usually provided by multiple firms, and hence construction documents do not have a uniform template. A title block is usually located on the lower right-hand corner of drawing documents but can be at the top, middle, or bottom of non-drawing documents. In addition, the shape and size of the title block are highly variable between different documents; therefore, an automated method should be capable of detecting the title block regardless of its location, shape, or size. Object detection API used to test title block detection approach.

The object detection API can find the location of the title block based on a pre-trained object detection model and extract the text of the title block by optical character recognition (OCR) engine. Then, the unstructured text will be used for document classification and information extraction. Document classification and information extraction are two independent sub-models that both use the text of the title block as their inputs. The dataset used to validate our method includes all construction documents, such as drawing documents, reports, and bill of materials. For the first experiment, 6000 construction documents from six classes were labelled, and the average precision was 98.8%. For the second experiment, 7000 construction documents from 32 classes were labelled, and the average precision was 91.7%. Experimental results on these 13000 construction documents demonstrate the effectiveness of using object detection API for title block detection. In the next step, the text of the title blocks was extracted by OCR techniques, and it was used for document classification and information extraction. The results show that using the text-only from the title block instead of the entire document increases the accuracy of the document classification and information extraction. The term frequency-inverse document frequency (TF-IDF) technique was used as text vectorization on the extracted text. Finally, the prepared dataset trained a Linear SVC classification model. The evaluation of the Linear SVC classification algorithm on 3200 documents showed an accuracy of 91.6%.

5.3.1 Proposed Methodology of title block detection

This section describes the four main stages steps of the proposed framework: (1) apply object detection API to detect the title block; (2) apply OCR on the detected block; (3) document classification; (4) extract information based on regular expressions. Figure 5-32 illustrates the proposed methodology for title block detection and information extraction.



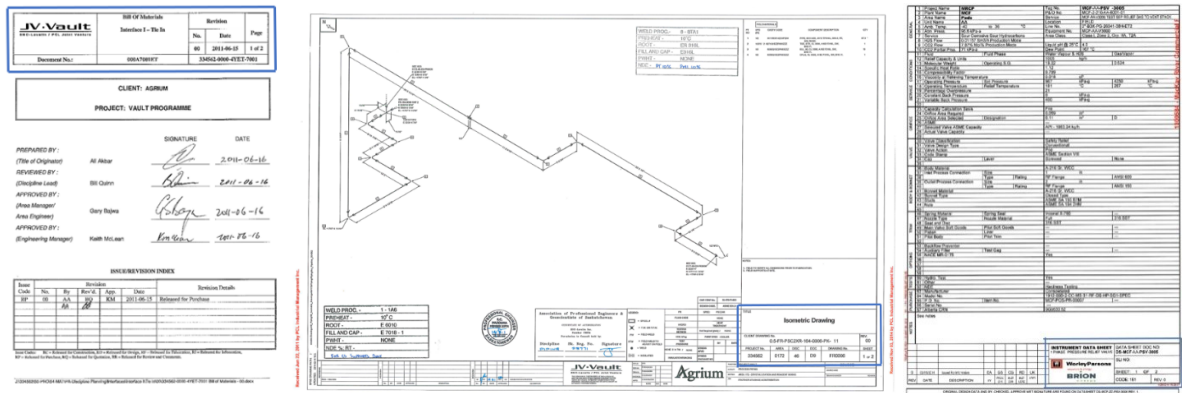
Figure 5-32. The proposed methodology of title block detection and information extraction

Object detection API

As applied to title blocks, object detection is the primary focus of this study, which is localizing and identifying an object or multiple objects in a single image. This work uses object detection to find the location of the title block, which contains information such as revision, document name, and document number. It can also find the relations between objects and provide a semantic description for them. The title block can be in different locations in different documents and can also have different sizes. Figure 5-33 shows a sample from our dataset where locations of the title blocks are shown using blue rectangles.

There are two widely used object detection APIs: Microsoft Azure Cloud object detection and Google TensorFlow object detection. In this study, TensorFlow object detection is selected, an open-source and cost-free API that can work on local machines [31]. This API provides a powerful object detection inference mechanism for recognizing and classifying objects within an image. It has four main parts: data preparation, feature extraction, building the classification model to classify extracted features, loading, and testing the classification model. Fast R-CNN, Faster R-CNN, you only look once (YOLO), and single-shot detector (SSD) are the most popular algorithms supported in the API [31]. Since accuracy is an important

factor in this study, Faster R-CNN was selected as it has the highest accuracy compared to the others [31]. As a result, TensorFlow object detection API implementation of the faster R-CNN detector [74] was used for object detection, while the LabelImg [128] image annotation tool was used to manually label the document images for training this model.



a) Bill of Materials

b) Isometric

c) Datasheet

Figure 5-33. Construction document example and location of the title block

OCR

Optical character recognition (OCR) refers to the image processing techniques for extracting text from scanned documents and images [129]. For this section, the Tesseract OCR library [130], one of the most accurate open-source OCR engines available, creates a text file for each document.

Document classification

TF-IDF (Term Frequency–Inverse Document Frequency) [131] is frequently used for text representation. TF-IDF can measure the importance of each term in a document. TF refers to the number of occurrences of a term in a document, and IDF is the ratio of the total number of documents to the number of documents that include the term [132]. Linear SVC [108] was used for document classification in the next

step. The Linear SVC implementation within Scikit-learn was used as an open-source machine learning library in Python [114].

Information extraction

In this study, information extraction is specified based on a pre-defined set of keywords and labels. For example, layout, isometric, map, BOM, and sheet are keywords used for document title extraction. The model searches through the text of the title block, looking for these keywords. Then, the model assigns the document title based on the extracted keywords that are found. For example, if the model finds “location” or “layout”, the document title is a layout diagram, which could be the same as document classification. The user will search for these keywords during manual classification to find the correct class, and the program will use the same approach. The keywords can either be determined by experts in the specific domain of the application or based on historical documents. Also, several pre-defined labels were used for information extraction, including revision and document number. The information extraction model searches the text for these labels to extract the next word. For example, if the text contains "Rev 2", the model will find "Rev" and will extract "2" as the revision number.

5.3.2 Experiments

Datasets

The original dataset had scanned documents in PDF format; however, object detection algorithms only accept images as inputs. Therefore, these PDF documents were converted to PNG and JPEG image formats.

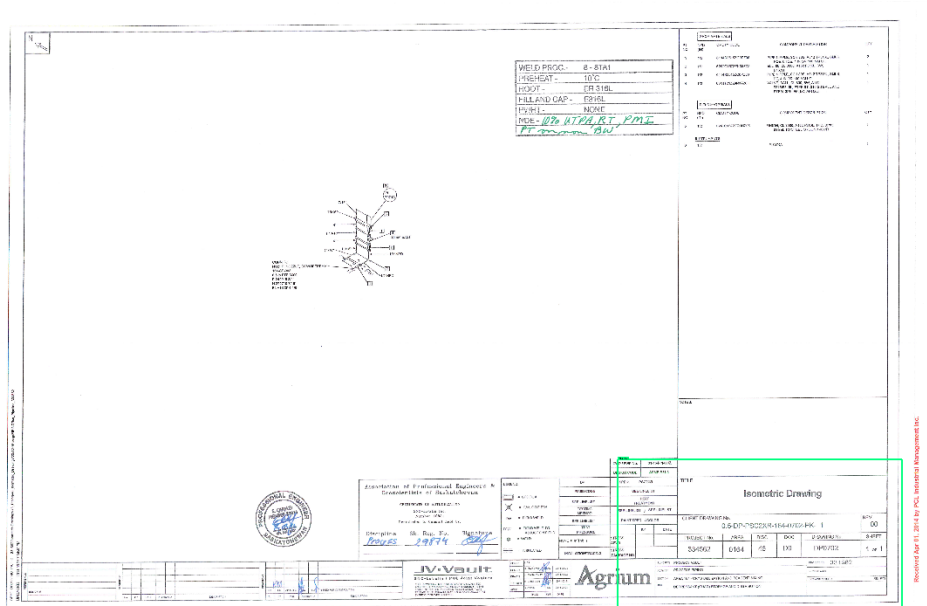
The provided datasets include a variety of image sizes, and, since we cannot define the model architecture without a fixed input size, I considered three options regarding the image sizes:

- 1- Keep the original size, then label and change the size during the training process

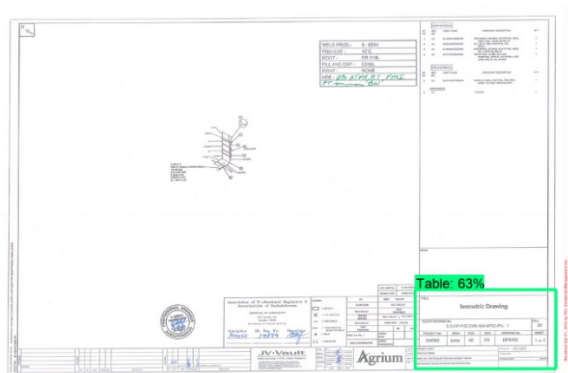
- 2- Resize the minimum dimension to 600 and maximum dimension to 1024 pixels resolution (which is the recommended size for the API), then label
- 3- Crop the images and keep only the segment containing the title block, then resize

The first set of experiments showed that the option 1 was successful for up to 2000 documents, but it failed during the training process when the number of documents increased to 4000. The third option was applicable for engineering drawings since the title block is usually located on the lower right-hand corner of these documents. However, it was not practical for non-drawing documents where the title block can be at the top, middle, or bottom of the document. Also, the accuracy of the second and third options was close for engineering drawings. As a result, the second option was adopted, as it was usable for both engineering drawing and non-drawing. The images were then resized to 600 * 1024 resolution, the recommended input size for the object detection API [74]. Figure 5-34 shows the results of all three options.

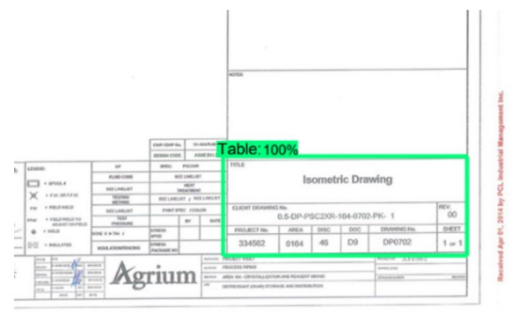
The proposed model was evaluated on two datasets. The first one includes 6000 construction documents evenly distributed between six document types. The second one has 7000 construction documents drawn randomly from 32 document types. The labelling, implementation, and training process are the same for both datasets. Only the number of document types is different. A comparison between the results for these two datasets is discussed in the section 5.3.4.



a) Keep the original size: 10200*6600



b) Resize the image: 1024*600



c) Crop the image: 512*300

Figure 5-34. Result of the three image resizing options

Labelling

LabelImg [128] is an open-source graphical annotation software used to annotate the document images with the location of the title blocks. This tool saves the locations as XML files that were further converted into the CSV format and then into TensorFlow record (TFRecord) for training in the API. 6000 construction

documents from the first dataset and 7000 construction documents from the second dataset were labelled manually. Figure 5-35 and Figure 5-36 illustrate manual labelling samples for different document types. The blue area is the title block. After manual labelling, a random 70 - 30 split was used to divide the dataset into training and validation parts for each dataset.

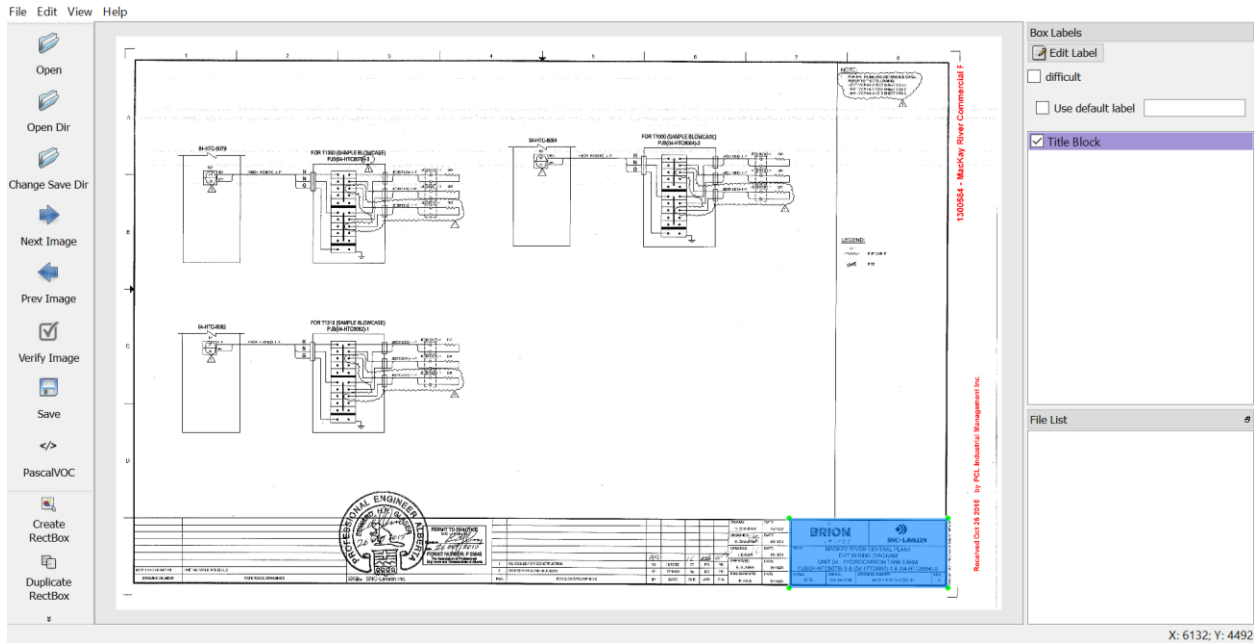


Figure 5-35. Sample of Manual labelling 1

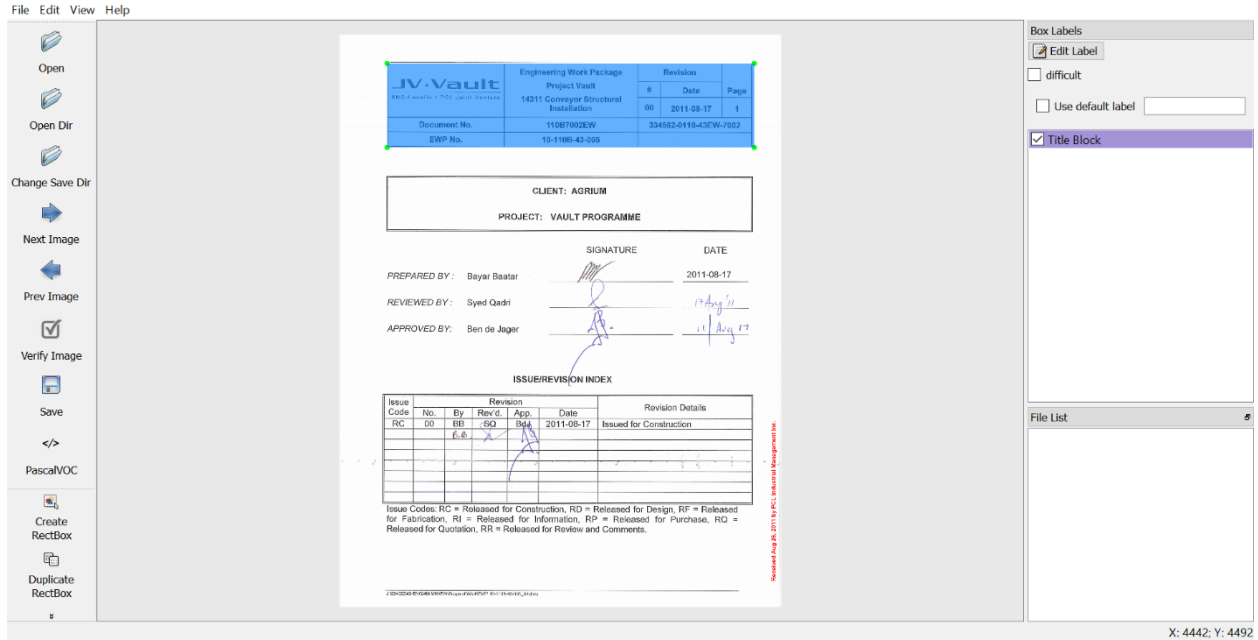


Figure 5-36. Sample of Manual labelling 2

5.3.3 Implementation and training

This work uses the TensorFlow object detection API implementation of the Faster R-CNN detector [34] with Inception_v2 backbone [133] pre-trained on the COCO dataset. Transfer learning was used to adapt this model to detect the title blocks by fine-tuning them on the labelled images of the construction documents [74]. Training and testing were done on an Ubuntu 16.04 system with 2 x GeForce GTX 1080 Ti GPUs and 64 GB RAM. Also, Tensorboard was used for monitoring and visualizing the training process. Figure 5-37 illustrates the steps of implementation and training, and Figure 5-38 shows sample results on Tensorboard, which were used to visualize the training process.

1. Label the images using the Labelling tool, which generates an XML annotation file for each image
2. Generate training and testing (validation) data:
 - a. Split the labelled images into test and training sets with a 30:70 ratio.
 - b. Convert all the training XML files into a single train CSV file (xml_to_csv.py)
 - c. Convert all the testing XML files into a single test CSV file (xml_to_csv.py)
 - d. Convert training CSV file into TFRecord (generate_tfrecord.py)
 - e. Convert testing CSV file into TFRecord (generate_tfrecord.py)
3. Create label map and configure training in (pipeline.py)
4. Select detection model: faster_rcnn_inception_v2
5. Configure the environment: faster_rcnn_inception_v2_pets.config
6. Train the model (train.py)
7. Export the model and generate the frozen inference graph containing the learned weights of the trained model
8. Evaluate the model based on the COCO metric (eval.py)

Figure 5-37. Data preparation and training steps

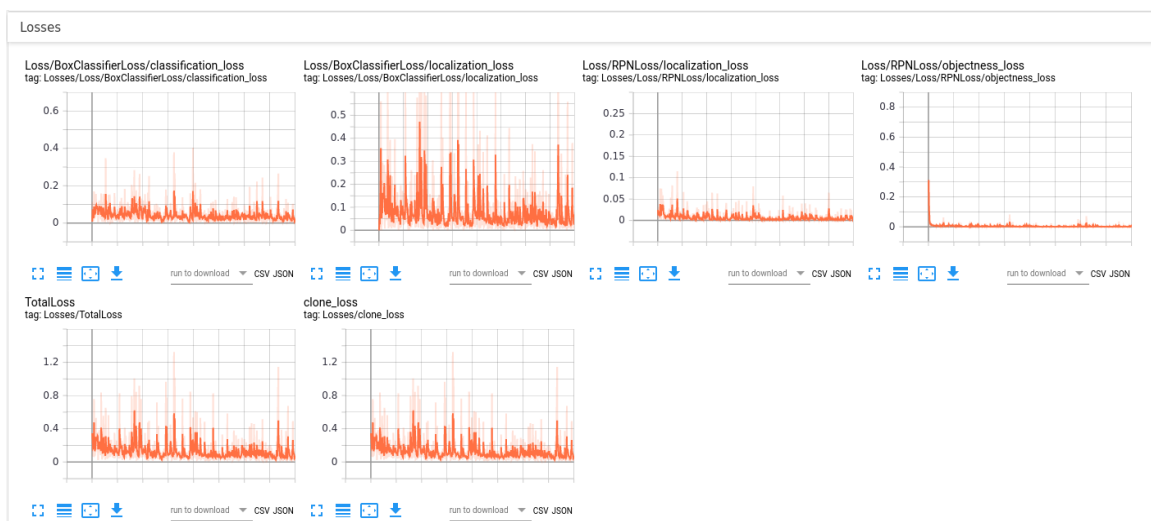


Figure 5-38. Example of result on Tensorboard

5.3.4 Evaluation and results

The output of the trained model is bounding boxes around the title blocks. Intersection over Union (IoU) is used to evaluate the output, which is an evaluation metric to measure the accuracy of an object detector. IoU is the overlap between the ground-truth bounding box and the predicted bounding box divided by the union area. IoU is defined as:

$$IoU = \frac{area(BB_{det} \cap BB_{gt})}{area(BB_{det} \cup BB_{gt})} \quad (5-5)$$

where (BB_{det}) and (BB_{gt}) are detected and ground-truth bounding boxes, respectively [134].

In this work, an IoU threshold of 0.5 was used; that is, if a detection achieved an IoU > 0.5, it was considered a successful detection, otherwise it was considered unsuccessful. Finally, the framework was evaluated on the title block detection task using the validation images comprising 30% of the labelled data. Figure 5-39 shows a sample of results on construction documents with green boxes showing the detected locations of the title blocks.

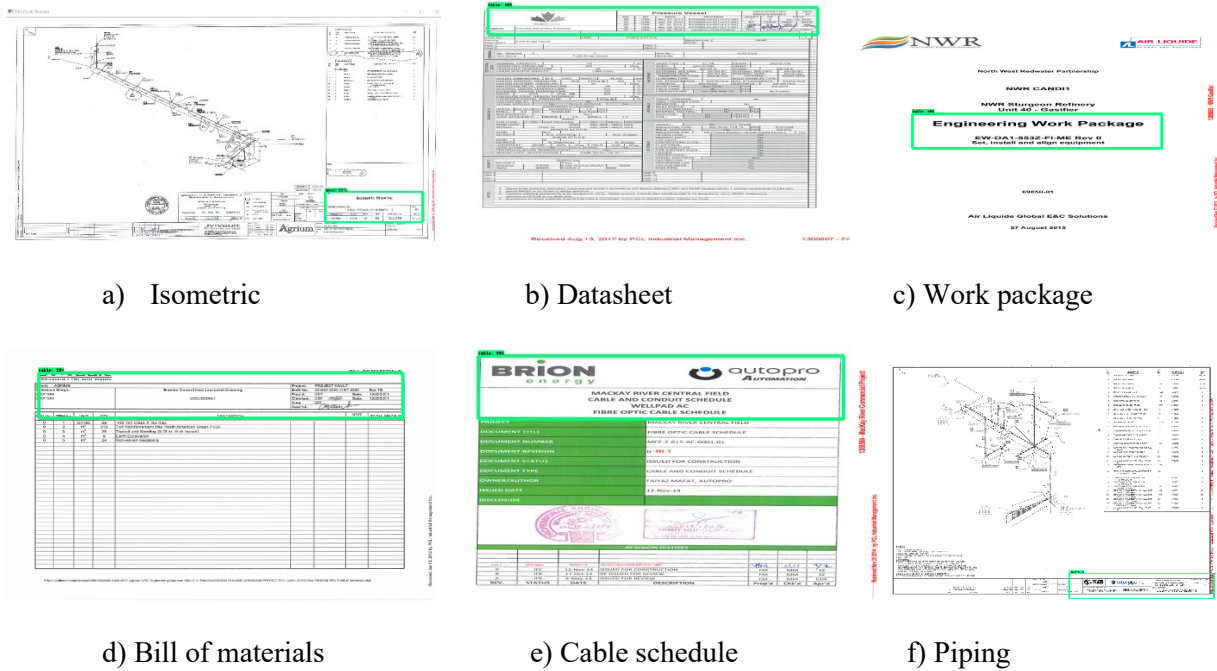
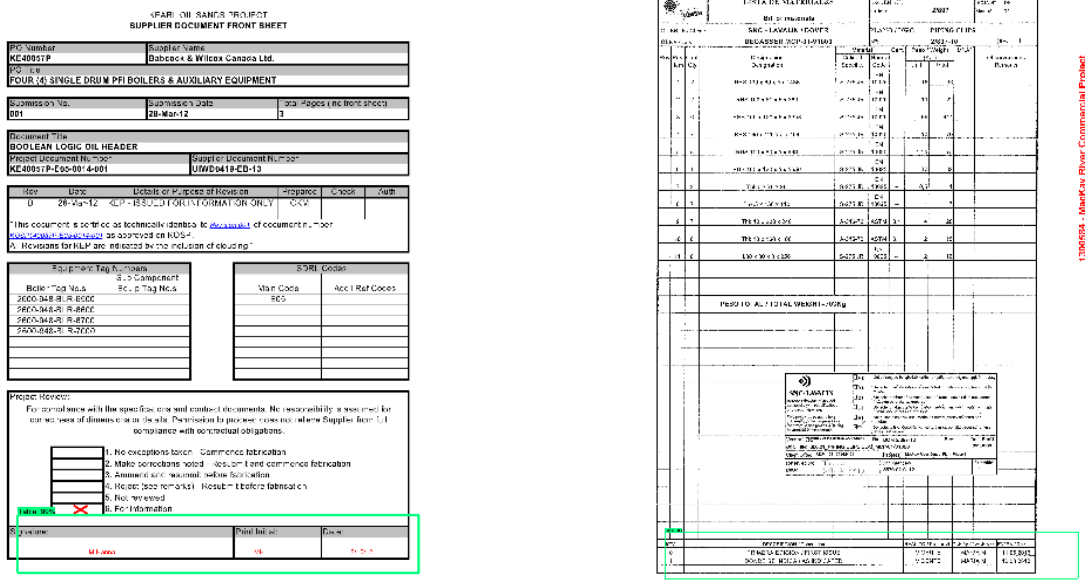


Figure 5-39. Sample of title block detection

Quantitative evaluation of the validation images gave an average precision of **98.8%** at 0.5 IoU for the first dataset and 91.7% for the second dataset. The second dataset has many more document types and, therefore, more layouts, decreasing precision. For the second dataset, 7000 construction documents were drawn randomly from 32 document types. Since each document type has more than one document layout, it seems likely that the object detection API was not well trained for all document types and document layouts. Figure 5-40 shows examples of title blocks that failed detections. For both examples, documents have several tables and complex structures. The title block is located on top of the page, while the title block detection selected the table at the bottom of the page. Table 5-20 shows the average precision of the proposed framework for both datasets. The high performance achieved shows its effectiveness.



a) Logic diagram

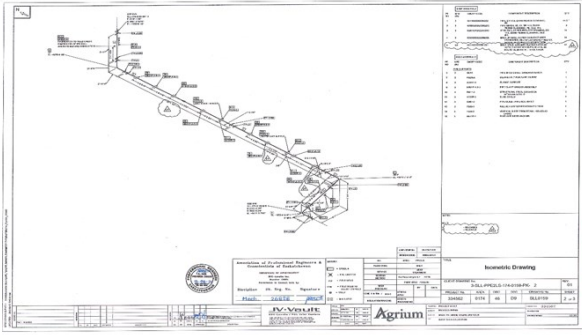
b) Bill of Materials

Figure 5-40. Sample results of failed title block detection

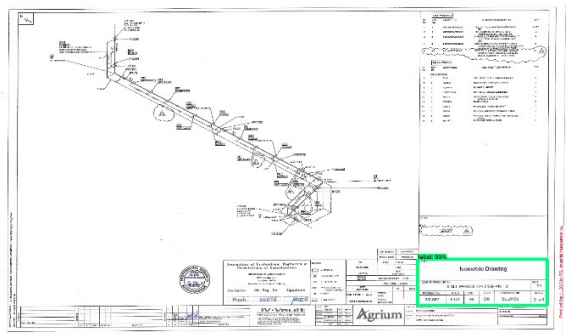
Table 5-20. Title block detection results

	Dataset 1	Dataset 2
Document numbers	6000	7000
Document types	6	32
Average precision	98.8%	91.7%

Figure 5-41 shows sample results for title block detection and information extraction. The model was able to assign document types and document numbers correctly. However, it could not find the document title and revision numbers. Quantitative testing on 3200 new construction documents gave our classification model an accuracy of 91.6% on unstructured documents. The classification model used TF-IDF and Linear SVC algorithms. However, the model is not reliable for extracting information such as revision and document number, which is related to the nature of our documents. For example, some documents do not have defined labels, or they use symbols like a triangle instead of "Revision" which OCR could not extract correctly. More information extraction methods need to be tested for increasing the accuracy of information extraction on unstructured documents.



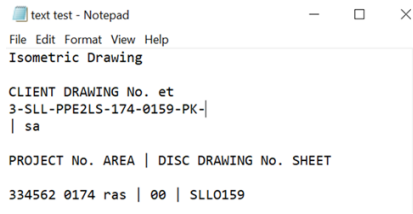
Input: Isometric Drawing



Detected table of information

TITLE					
Isometric Drawing					
CLIENT DRAWING No. 3-SLL-PPE2LS-174-0159-PK- 2					REV. 01
PROJECT No.	AREA	DISC	DOC	DRAWING No.	SHEET
334562	0174	46	D9	SLL0159	2 of 3

Image crop of the table of information



Text extraction

Document type: isometric
Drawing No: 3-sll-ppe2ls-174-0159-pk

Information extraction

Figure 5-41. Sample results of title block detection and information extraction

Chapter 6 Test of Title Block Detection and Information Extraction

Model

6.1 Test of title block detection on drawings documents

The title block has essential information about engineering drawings, including the document name, document number, revision number, job number, and scale. The primary step in the document storing process is identifying the document information and extracting the information, which includes many manual tasks. Researchers presented several methods for automated table detection and recognition. In this research, TensorFlow object detection API and Faster R-CNN model were applied on drawing documents to estimate the location of the title block. Five tests were designed for title block detection on drawing documents; the number of documents, document types, number of labels, and size of documents were changed for the following tests.

Test 1

The title block is a table including several text boxes, such as title, revision number, document number, project name, date, company name, etc. Test 1 was designed to evaluate the ability of text box recognition, such as the “revision number” text box, which is part of the title block. Four classes were selected for this test: Isometric Drawings, Layout Drawings, Schematic Drawings, and Wiring Drawings. Each drawing was labelled based on four attributes: drawing type, title, number, and revision. The sample dataset contains 4000 images with 1000 images of each class.

Table 6-1. The architecture of Test 1

Number of Documents	4000
Number of document type	4
Number of labels	4
Size of Documents	Original size
learning rate	1.999

The trained model failed to recognize all the annotated areas indicated during the labelling phase.

The model was not able to recognize each text box. Therefore, I designed the test to examine whether the model can detect the whole title block instead of each text box.

Test 2

The purpose of Test 2 is to evaluate the ability of title block detection. For Test 2, only the title block is labelled, and similar to Test 1, four classes were selected: Isometric Drawings, Layout Drawings, Schematic Drawings, and Wiring Drawings. The sample dataset was taken as 4000 images with 1000 images of each class. Each drawing was labelled based on the title block.

Table 6-2. The architecture of Test 2

Number of Documents	4000
Number of document type	4
Number of labels	1
Size of Documents	Original size
Initial learning rate	0.0002
Number of steps	4600

The trained model failed while the system caused several out-of-memory and memory allocation errors.

Therefore, I decreased the amount of data to test the model again.

Test 3

Test 3 is designed the same as Test 2 with less amount of data. The sample dataset was taken as 1000 images of isometric drawings. Each drawing was labelled based on the title block.

Table 6-3. The architecture of Test 3

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	Original size
Initial learning rate	0.0002
Number of steps	4500

Test 3 successfully detects the title block on isometric drawing, and test evaluation shows average precision at 0.5 IoU: 0.77. Figure 6-1 illustrates the result of Test 3.

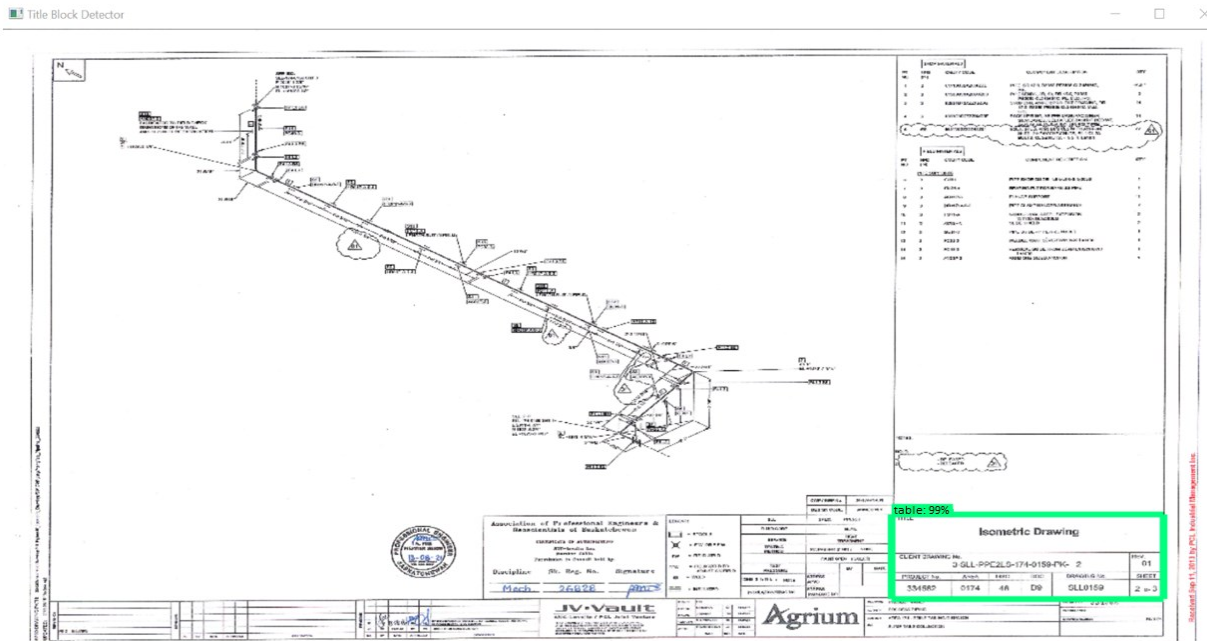


Figure 6-1. Result of Test 3

Test 3 has two challenges: Challenge 1: The model shows more than one detected bounding box for some images. Figure 6-2 illustrates the example of the challenge.

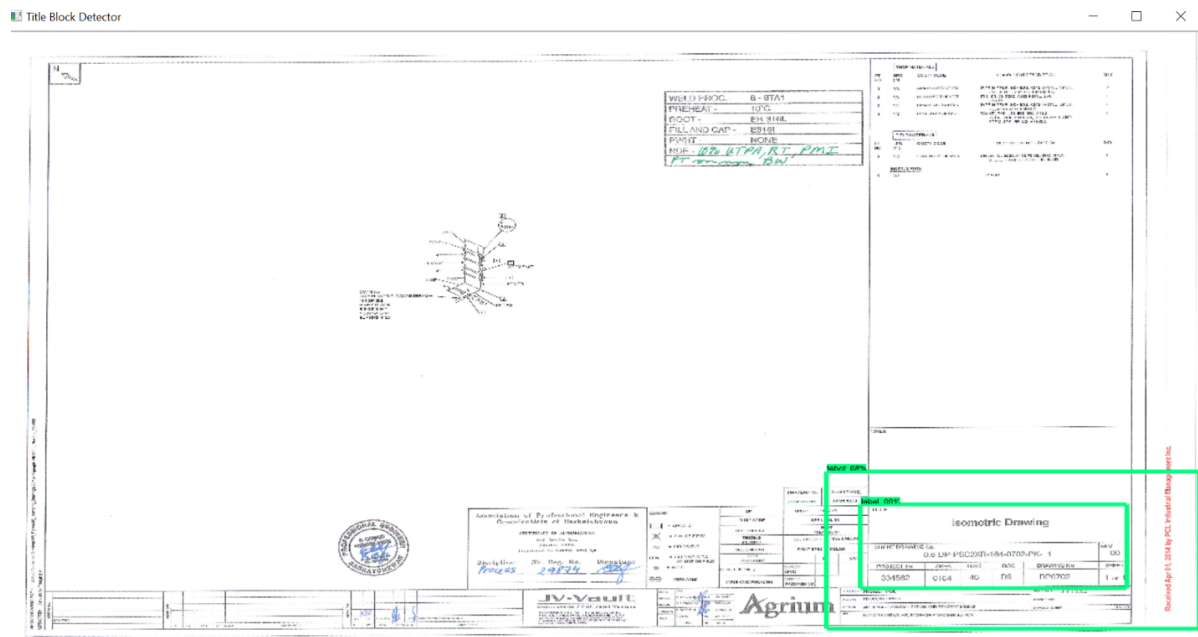


Figure 6-2. Sample of result that has two detected bounding boxes

Challenge 2: for noisy images, it cannot show any detected bounding box. Figure 6-3 illustrates the noisy documents without a bounding box.

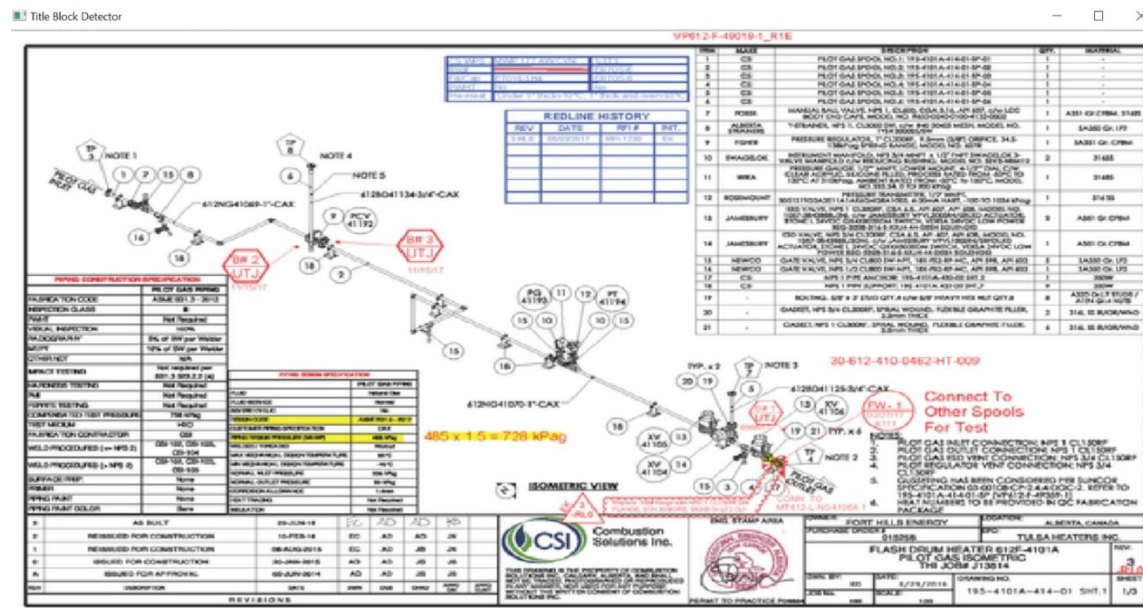


Figure 6-3. Sample of the noisy image

Test 3 was repeated for three more document types: Layout, schematic, and wiring. The average precision at 0.5 IoU was between 0.85 to 0.77. Figure 6-4 and Figure 6-5 illustrate the result of this test.

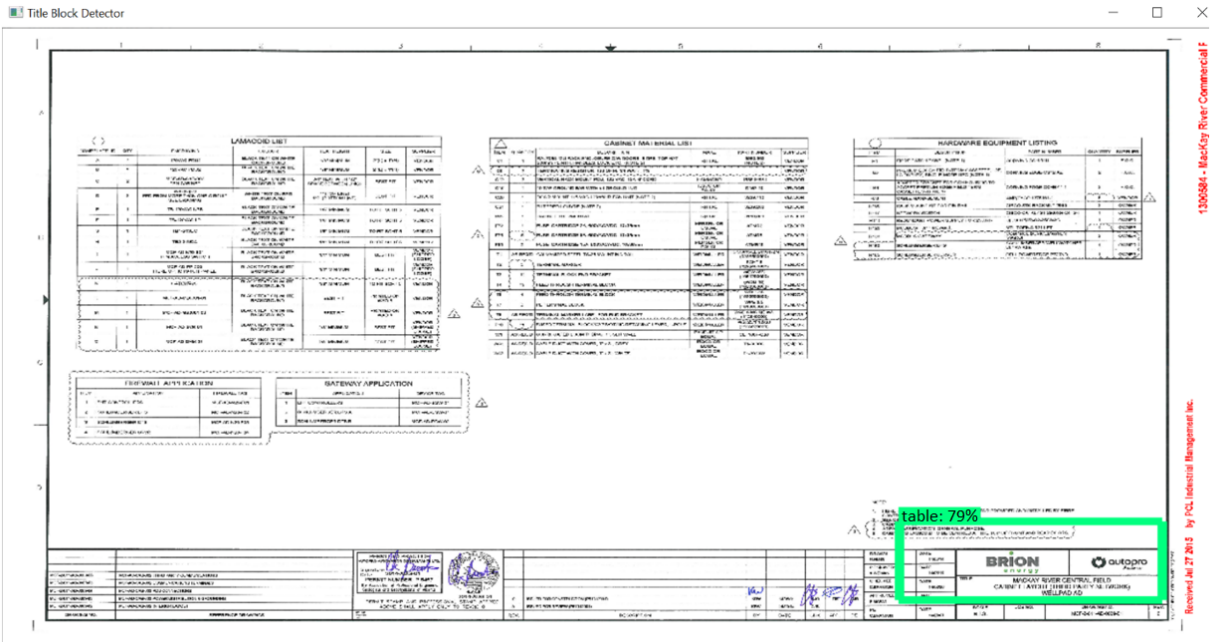


Figure 6-4. Layout Drawing

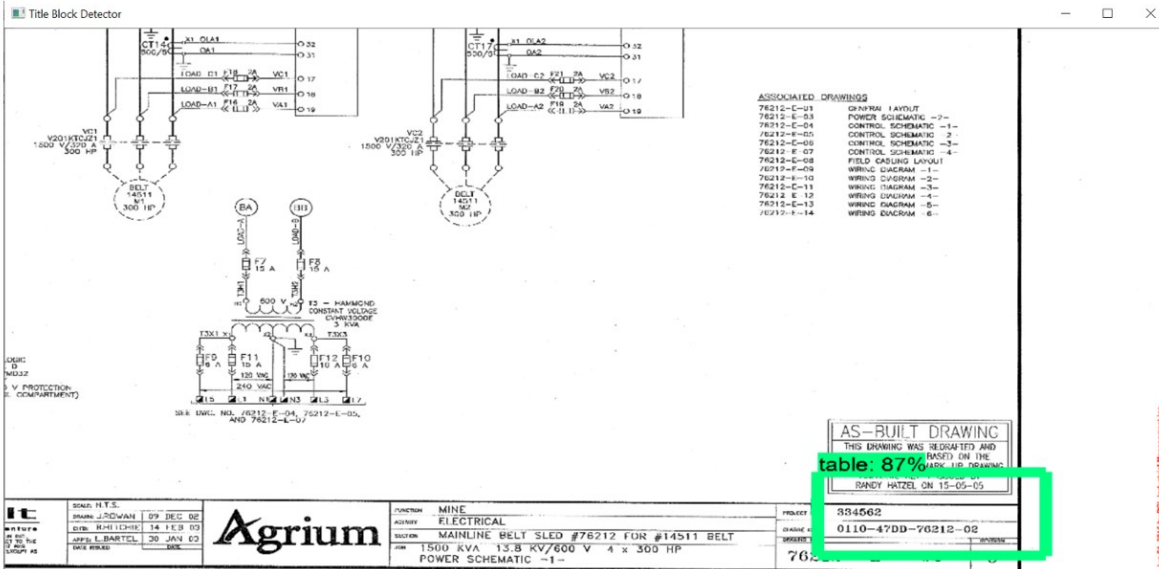


Figure 6-5. Schematic Drawing

Test 4

Test 4 is designed to evaluate the title block detection on split documents. The sample dataset was taken as 1000 images of isometric drawings. Each drawing was labelled based on the title block. The size of the document was changed into ¼ original documents by the program in Matlab, and only the image with the title block was used for labelling and training purposes.

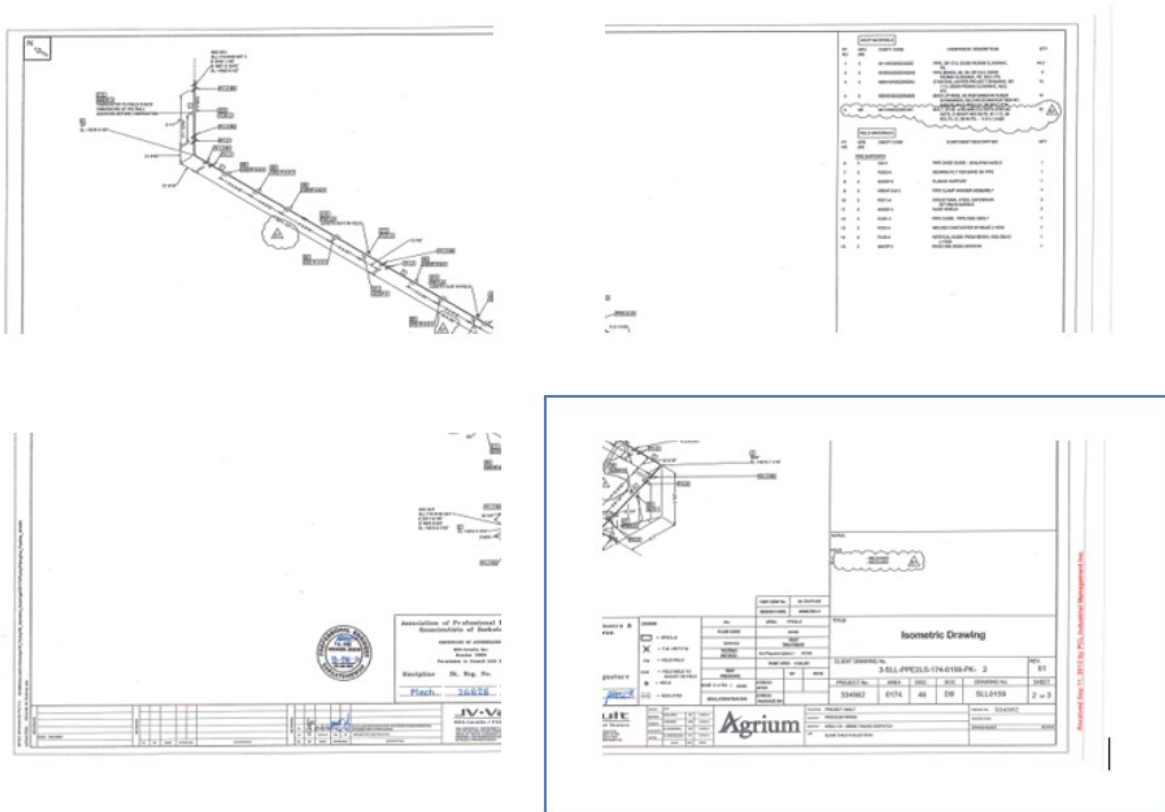


Figure 6-6. Sample of divided images

Table 6-4. The architecture of Test 4

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	Split into 4 parts
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.99, which means the accuracy is close to 100%.

Figure 6-7 shows the sample of the result.

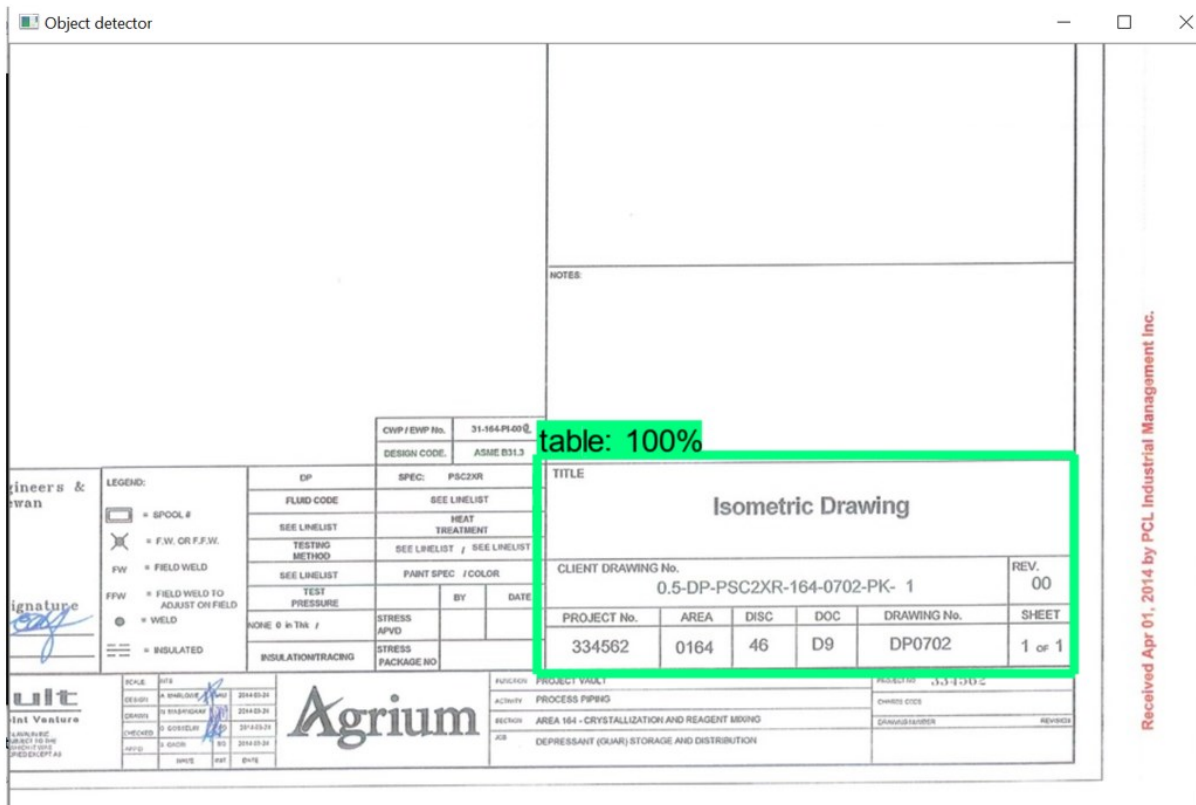


Figure 6-7. Result of Test 4 – Isometric Diagram

Test 4 was repeated for three more document types: Layout Drawings, Schematic Drawings, and Wiring Drawings. The average precision at 0.5 IoU was between 0.95 to 0.99.

In addition, Test 4 was repeated as the combination of 4000 data and four document types. The average precision at 0.5 IoU was 0.99. Figure 6-8, Figure 6-10 and Figure 6-10 are the samples of results.

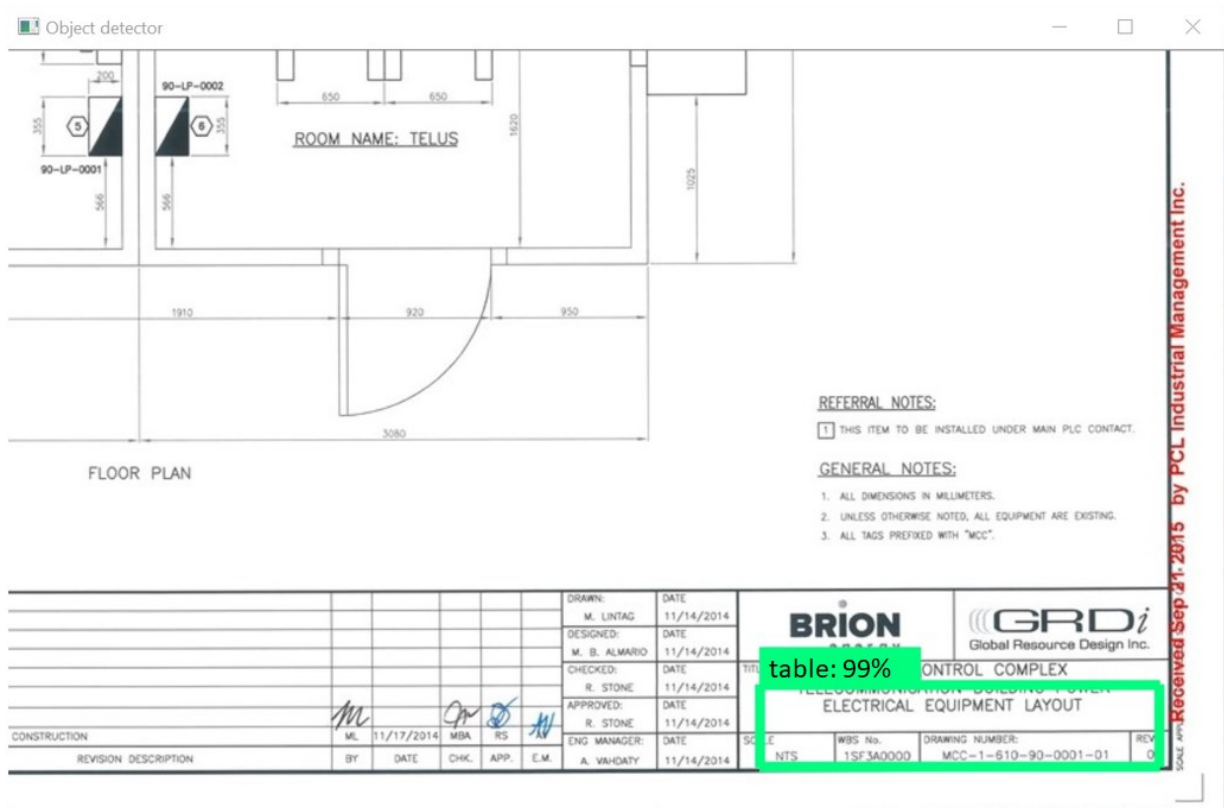


Figure 6-8. Result of Test 4 - Layout

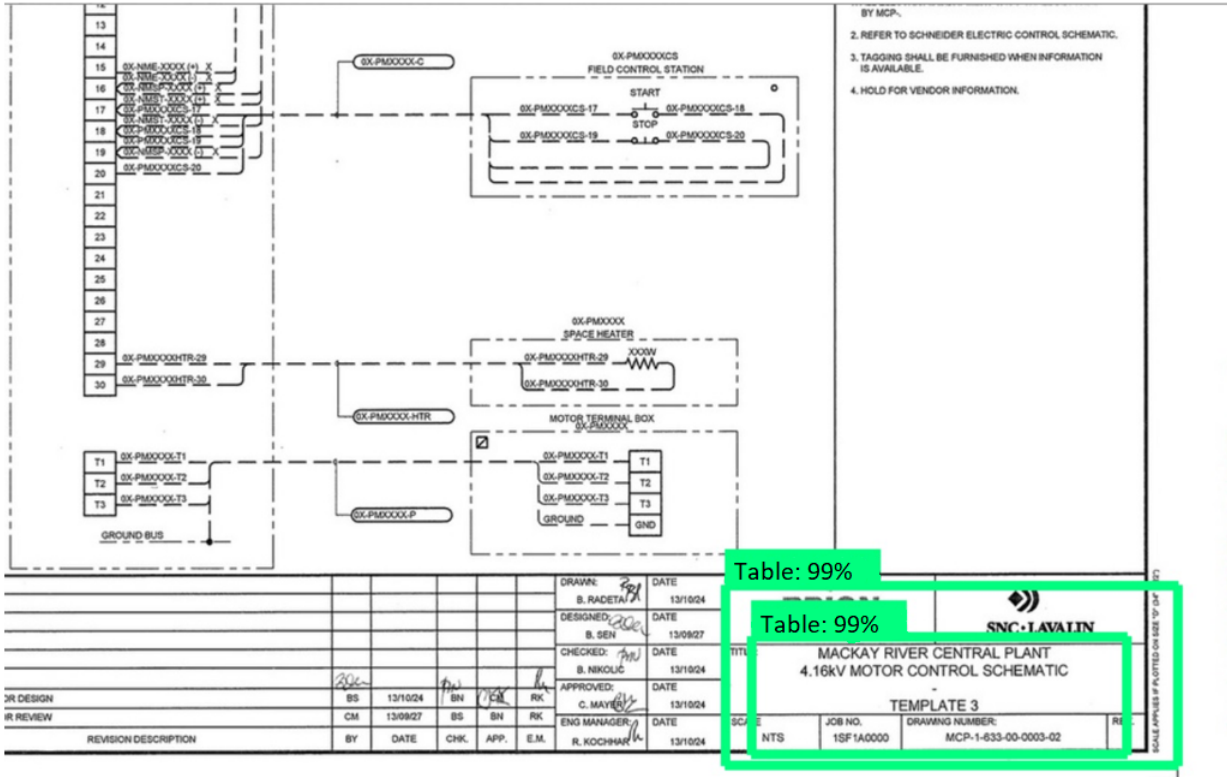


Figure 6-9. Result of Test 4 - Schematic

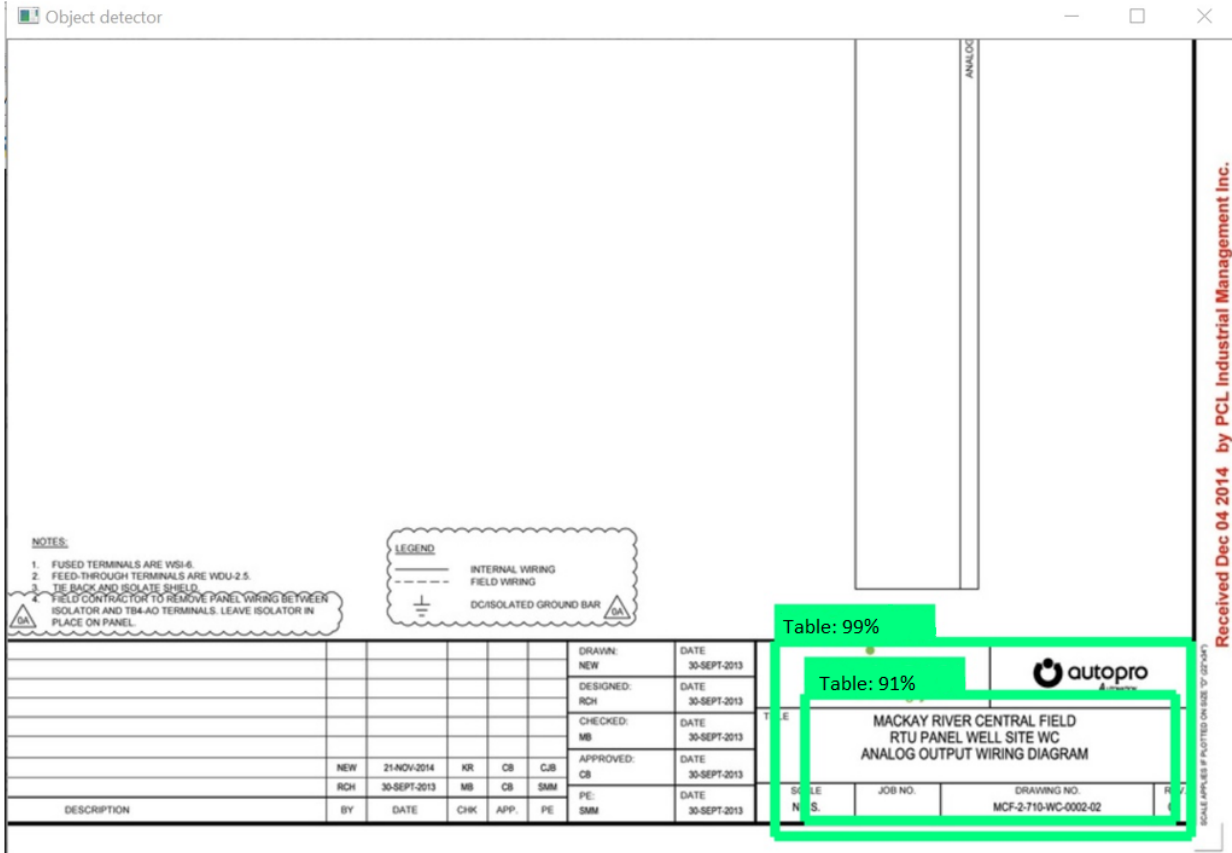


Figure 6-10. Result of Test 4 - Wiring Diagram

Test 5

Test 5 is designed to evaluate the title block detection on 600 *1024 documents size. The dataset of this test has 1000 Isometric drawings, and the size of the drawing changed to 600 *1024, which Faster R-CNN recommends.

Table 6-5. The architecture of Test 5

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	600 *1024
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.9267

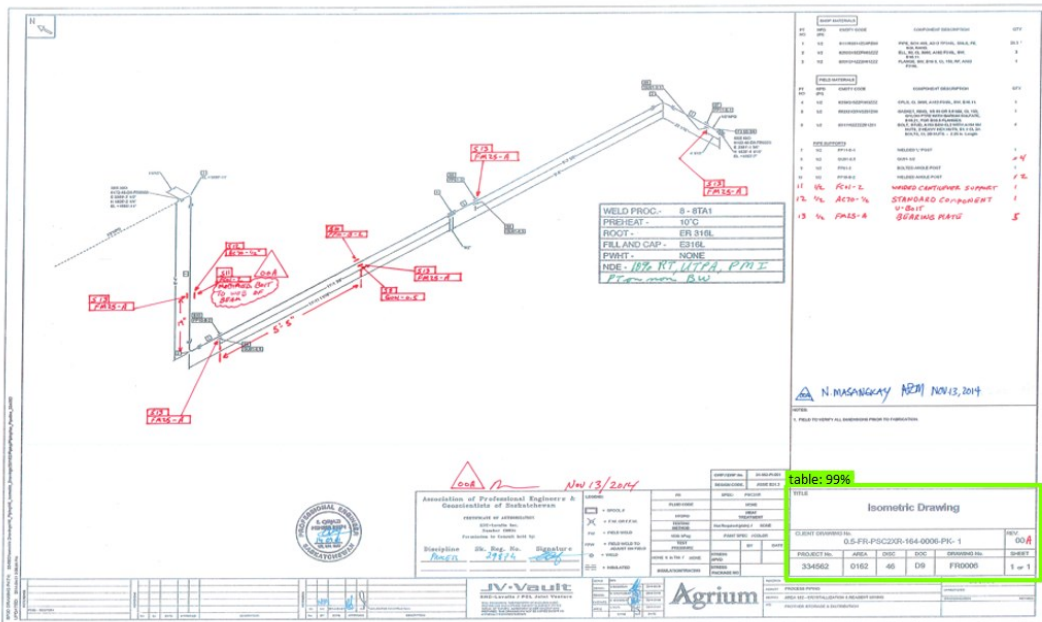


Figure 6-11. Result of Test 5- Isometric drawing-A)

Test 5 was repeated for three more document types: layout drawings, schematic drawings, and wiring drawings. The average precision at 0.5 IoU was between 0.92 to 0.96.

Test 5 was repeated as the combination of 4000 data and four document types. The average precision at 0.5 IoU was 0.97. Figure 6-12 shows the sample of the result.

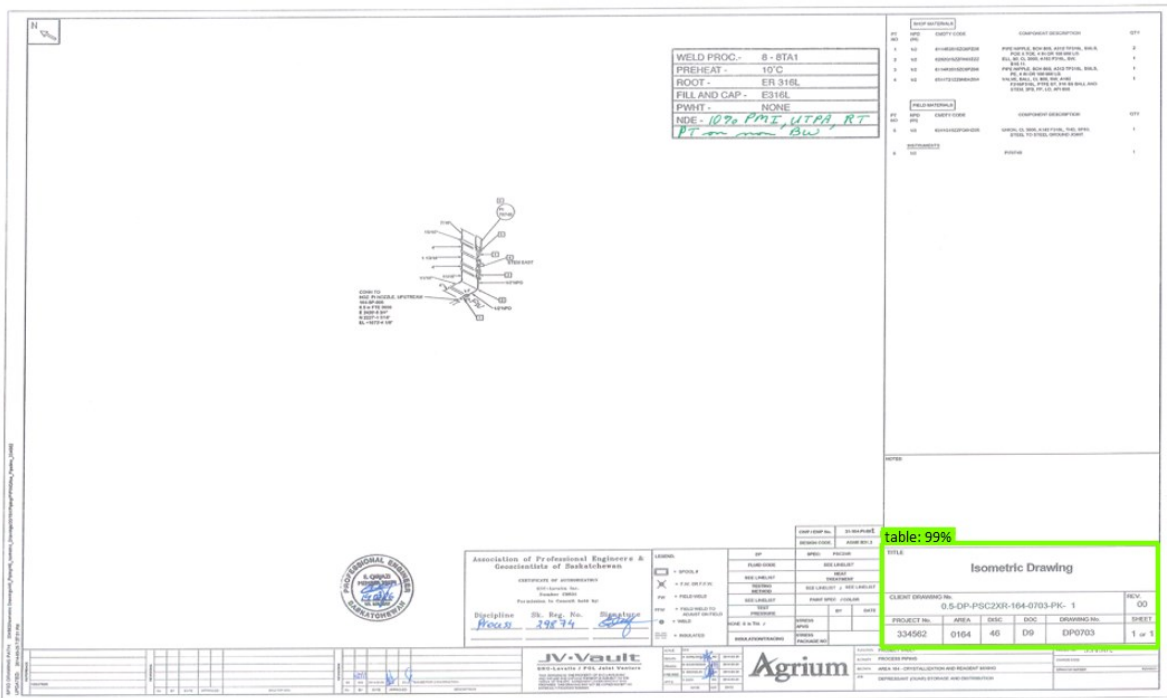


Figure 6-12. Result of Test 5- Isometric drawing- B)

The result of tests 4 and 5 show that the split and reduced size of the document have similar accuracy. At the same time, the reduced size of the document is applicable to all types of documents. Table 6-6 illustrates the summary of results for title block detection on drawing documents.

Table 6-6. Summary of Results for title block detection on drawing documents

Test no	Number of Documents	Number of document type	Document type	Number of labels	Size of Documents	average precision at 0.5 IoU
1	4000	4	Isometric- Layout- Schematic- Wiring	4	Original	fail
2	4000	4	Isometric- Layout- Schematic- Wiring	1	Original	fail
3	1000	1	Isometric	1	Original	0.77
3	1000	1	Layout	1	Original	0.85
3	1000	1	Schematic	1	Original	0.80
3	1000	1	Wiring	1	Original	0.84
4	1000	1	Isometric	1	Split	0.99
4	1000	1	Layout	1	Split	0.99
4	1000	1	Schematic	1	Split	0.97
4	1000	1	Wiring	1	Split	0.95
4	4000	4	Isometric- Layout- Schematic- Wiring	1	split	0.99
5	1000	1	Isometric	1	Reduce	0.92
5	1000	1	Layout	1	Reduce	0.96
5	1000	1	Schematic	1	Reduce	0.94
5	1000	1	Wiring	1	Reduce	0.96
5	4000	4	Isometric- Layout- Schematic- Wiring	1	reduce	0.97

6.2 Test of title block detection on non-drawings documents

Test of title block detection on non-drawing documents was the same process of drawing documents; Title block can be in a different location at the non-drawing document. Therefore, the split option was not practicable for non-drawing documents. Also, the original document size was not practicable due to the memory size error. Therefore, only reduced document size was considered for this test. Five tests were designed for title block detection on non-drawing documents, the number of documents and document types were changed for the following tests.

Test 1

Test 1 is designed to evaluate the title block detection on non-drawing documents, such as bill of materials, work package, and cable schedule. The dataset size is 1000 Datasheet document, and the size of the document changed to 600 *1024, which is recommended by Faster R-CNN.

Table 6-7. The architecture of Test 1

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	600 *1024
Initial learning rate	0.0002
Number of steps	200000

The result of Test 1 indicates that the model successfully identified the location of the title block on the datasheet. Test evaluation shows average precision at 0.5 IoU: 0.993

Table 994

Pressure Vessel		DATA SHEET NO.		REV
NO.	BY	DATE	REVISION	DATE
20	TRK	NOV 07 2013	REVISED DOTTLEPFEIST	2013.11.07
21	TRK	JAN 24 2014	REVISED DOTTLEPFEIST	2014.01.24
22	TRK	JUN 10 2014	REVISED DOTTLEPFEIST	2014.06.10
23	TRK	MAR 04 2015	REVISED DOTTLEPFEIST	2015.03.04
24	TRK	JUN 26 2015	REVISED DOTTLEPFEIST	2015.06.26

PROJECT: Fort Hill Secondary Extraction

ASSET No.: PABO PAB12A-013-2

Manufacturer: ORME

Item No.: 6120-2000

Item Name: Froth Surge Vessel

Normal Capacity: 170

Operating Pressure: 800

Operating Temperature: 300

Liquid Specific Gravity: 1.000 (max)

Vessel Dimensions: ID: 3600, TD: 16450

Design Internal Pressure: 2000

Design External Pressure: 0

Min Design Metal Temperature: 0

Design Temperature: 300

Max Design Temperature: 300

Pressure Class: 200

Pressure Group: 200

Pressure Category: 200

Design Code / Specification: ASME Section VIII Div.1

Weight: 22700

Operating Weight (Startup Ready): 30000

Empty Weight: 25000

Empty Weight: 25000

Notes:

- Vessel to be designed, fabricated, inspected and tested in accordance with Surcor Standard 0601 and ASME Section VIII Div.1 and the requirements of CGA B51.
- Vessel MANVO to be fitted to design conditions.
- Pressure retaining components to be designed to 20°C. Vessel support, external clips and lifting lugs to be designed to -25°C, ASMT temperature reduction in ASME is not applicable.
- All pressure retaining materials shall be fully killed & normalized. Impact test is mandatory when required per Code.

Received Aug 13, 2017 by PCL Industrial Management Inc.

1300607 - FH

Figure 6-13. Result of Test 1- Datasheet

Test 2

Test 2 is designed to evaluate the title block detection on non-drawing documents. The size of the dataset is 1000 Work Package documents. The size of the document changed to 600 * 1024, which Faster R-CNN recommends. Table 6-8 shows the architecture of the test.

Table 6-8. The architecture of Test 2

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	600 * 1024
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.855. Figure 6-14 shows the sample of results.



North West Redwater Partnership

NWR CANDI1

NWR Sturgeon Refinery
Unit 40 - Gasifier

Table: 99%

Engineering Work Package

EW-DA1-553Z-FI-ME Rev 0
Set, install and align equipment

69850-01

Air Liquide Global E&C Solutions

27 August 2015

1300605 - NWR Gasifier

Received Sep 22 2015 by PCL Industrial Management Inc.

Figure 6-14. Result of Test 2- work package

Test 3

Test 3 is designed to evaluate the title block detection on non-drawing documents. The size of the dataset is 1000 Bill of Material documents. The size of the document changed to 600 *1024, which Faster R-CNN recommends. Table 6-9 shows the architecture of the test.

Table 6-9. The architecture of Test 3

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	600 *1024
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.995. Figure 6-15 shows the sample of results.

Table: 99%

REV. No.	ITEM No.	UNIT	QTY	DESCRIPTION	UNIT	TOTAL PRICE (\$)
0	1	tonnes	29	400 mm Class A Rip Rap		
0	2	m ²	510	Turf Reinforcement Mat, North American Green P550		
0	3	m ³	28	Topsoil and Seeding (0.05 m. thick topsoil)		
0	4	m ²	9	Earth Excavation		
0	5	m ²	24	Non-woven Geotextile		

Client: AGRIMUM		Erosion Control Near Low Level Crossing		Project: PROJECT VAULT	
Reference Dwg: DA1289		202C8069ET		BoM No: 334562-0202-41ET-8069	Rev PB
DA1290				Prep'd: DDT	Date: 12/22/2011
				Checked: CSF	Date: 12/22/2011
				Area: 202	
				Appr'd: <i>[Signature]</i>	

\\sas1-s-filesrv1\data\Active\2008\1362\08-1362-0571 Agrimum VPO Engineering Services TMA 5 Yr Plan\DIVERSION CHANNEL\EROSION PROTECTION LOW LEVEL\Rev PB\BOM REV 0 Bill of materials.xlsx

Received Jan 18, 2012 by PCL Industrial Management Inc.

Figure 6-15. Result of Test 3- Bill of Materials

Test 4

Test 4 is designed to evaluate the title block detection on non-drawing documents. The size of the dataset is 1000 Cable Schedule documents. The size of the document changed to 600 *1024, which Faster R-CNN recommends. Table 6-10 shows the architecture of the test.

Table 6-10. The architecture of Test 4

Number of Documents	1000
Number of document type	1
Number of labels	1
Size of Documents	600 *1024
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.962. Figure 6-16 shows the sample of results.

Table: 99%

PROJECT	MACKAY RIVER CENTRAL FIELD
DOCUMENT TITLE	FIBRE OPTIC CABLE SCHEDULE
DOCUMENT NUMBER	MCF-2-615-AC-0001-01
DOCUMENT REVISION	0-RL1
DOCUMENT STATUS	ISSUED FOR CONSTRUCTION
DOCUMENT TYPE	CABLE AND CONDUIT SCHEDULE
OWNER/AUTHOR	FAIYAZ MAFAT, AUTOPRO
ISSUED DATE	12-Nov-14
DISCLOSURE	

REV.	STATUS	DATE	DESCRIPTION	Prep'd	Chk'd	Apr'd
0-RL1	DECLINED	30-MAY-10	REF: PCL-MCP-02524 (REF: SITE-1486)			
0	IFC	12-Nov-14	ISSUED FOR CONSTRUCTION	FM	MM	YZ
B	IFR	17-Oct-14	RE-ISSUED FOR REVIEW	FM	MM	YZ
A	IFB	9-May-14	ISSUED FOR REVIEW	FM	MM	CDB

1300534- Mackay River Commercial Project

Received Nov 25 2014 by PCL Industrial Management Inc.

Figure 6-16. Result of Test 4- cable schedule

Test 5


Test 5 is designed to evaluate the title block detection on non-drawing documents. The dataset size is increased to 4000 documents: 1000 Datasheet, 1000 Work Package, 1000 Bill of Material, and 1000 Cable Schedule documents. The size of the document changed to 600 * 1024, which Faster R-CNN recommends. Table 6-11 shows the architecture of the test.

Table 6-11. The architecture of Test 5

Number of Documents	4000
Number of document type	4
Number of labels	1
Size of Documents	600 * 1024
Initial learning rate	0.0002
Number of steps	200000

Test evaluation shows average precision at 0.5 IoU: 0.92. Figure 6-17 and Figure 6-18 show the sample of results.

Table: 994

		Fort Hills Secondary Extraction Project WBS No: 03-00114-FT-SE Folder No: Contractor:							
CONSTRUCTION WORK PACKAGE									
CWP	Plant	Discipline	Model/Work Area	Type	Sequence No.				
CWP	612	P	2 0	1	S	B	0	0	1
Description: Power transformer and NGR installation (CWA11-2 E-House #2)									
Rev. No.	Revision Description	Date	Originator	Checked by:	Approved by:				
1A	Issued for Construction	Aug 25, 2015	CC	RL PC	PE	RCL	CM		
1B	Issued for Construction	Nov 20, 2015	Yes	JBL BK CK	ADM	SM	KB/SR		

Legend: Construction Coordinator(CC), Originator (ORIG), Central Construction Planning Manager (CCPM), Construction Manager (CM), Project Engineer (PE), Regulatory Lead (RL), Regulatory Compliance Lead (RCL), Project Controls (PC)

This CWP has been revised (Rev.1B) as noted herein.

CWP Revision History		
Rev.#	Section Changed	Revisions Made
1B	3.1.1 4	Added Scope of Work – Chipping & Grouting for Transformer Foundations
1B	3.2.1	Updated Engineering Drawing List
1B	3.2.2	Updated Vendor Drawings and Documents
1B	3.2.3	Removed Reference Documents

100607-FHSE - Above Ground

Received Nov 20 2015 by PC, Industrial Management Inc.

Figure 6-17. Result of Test 5- work package

Description		Units	Design Data	Rev
1	GENERAL BRIBER DATA			
2	EQUIPMENT NUMBER/TAGS:		170-SG-001 & 172-SG-001	
3	VENDOR		POWELL ELECTRICAL SYSTEM	
4	VENDOR QUOTATION NUMBER		80068-806	
5	QUOTATION DATE			
6	MANUFACTURER		POWELL ELECTRICAL SYSTEM	
7	MODEL DESIGNATION or NUMBER		POW VAG 104	
8	REFERENCE (DRAWING(S))		SINGLE LINE DIAGRAM 80067000 & 80067007	
9				
10	AMBIENT ENVIRONMENTAL CONDITIONS			
11	ELEVATION ABOVE SEA LEVEL	m	REFER TO SPECIFICATION 334552-0000-41EG-7001	
12	EQUIPMENT LOCATION (INDOOR/OUTDOOR)		INDOOR(S)	
13	UNUSUAL CONDITIONS (SUNSHINE/SHADOWS)		POTASH DUST	
14	HAZARDOUS AREA CLASSIFICATION		UNCLASSIFIED	
15	AMBIENT DESIGN TEMPERATURE RANGE	°C	-30 TO +40 °C	
16	SEISMO CLASSIFICATION		REFER TO SPECIFICATION 334552-0000-41EG-7001	
17				
18	ELECTRICAL SYSTEM PARAMETERS			
19	NOMINAL SUPPLY VOLTAGE	kV	13.8 kV	
20	SUPPLY FREQUENCY	Hz	60 Hz, +/-0.5 Hz	
21	SYSTEM MAXIMUM SHORT CIRCUIT LEVEL	KA	50	
22	SUPPLY SYSTEM GROUNDING		500A, 10 Sec (RGR Grounded)	
23	SYSTEM DESIGN LIFE	years	50	
24				
25	ELECTRICAL DESIGN CONDITIONS			
26	NOMINAL VOLTAGE CLASS	kV	18 kV	
27	NOMINAL MVA CLASS	MVA	1000 MVA	
28	RESISTANCE/INDUCTIVE TYPE (S/C/B/LOW)		LOW RESISTANCE GROUNDING 500A, 10Sec	
29	ENCLOSURE: REMANENTIA TYPE		AR TYPE 2B	
30	OUTDOOR (WALK-IN, NON WALK-IN)		NOT APPLICABLE	
31	ARC PROOF		PROVIDED	
32	EQUIPMENT IEL RATINGS	kV	38 kV	
33	TYPICAL LEAKAGE		NOT REQUIRED	
34	VERMIN PROOFING		PROVIDED	
35				
36	BUS DATA			
37	CONTINUOUS CURRENT RATING	A	8000A FOR 170-SG-001 & 1000A FOR 172-SG-001	
38	SHORT CIRCUIT RATING	KA	50kA/10Sec (10Sec)	
39	BUS MATERIAL		COPPER, T3N PLATED	
40	INSULATION		EPoxy	
41	BUS HOT SPOT TEMPERATURE	°C	65	
42	GROUNDED BUS SIZE, MATERIAL	mm	1/2" x 6" - COPPER	
43				
44	CURRENT BREAKER		VACUUM	
45	TYPE (AIR, SF6, VACUUM)		TWO HORN	
46	BREAKER MOUNTING CONFIGURATION ONE/TWO HORN		50kA/10Sec (10Sec)	
47	MINIMUM VOLTAGE FOR RATED BREAKER MVA	kV	50kV ALL VOLTAGES	
48	LINE FREQUENCY WITHSTAND (MIN.)	kV	50kV	
49	RATED CURRENT - MAIN BREAKER	A	8000A FOR 170-SG-001 & 1000A FOR 172-SG-001	
50	RATED CURRENT - THE BREAKER	A	8000A FOR 170-SG-001 & 1000A FOR 172-SG-001	

Received Jun 26, 2015 by PC, Industrial Management Inc.

Figure 6-18. Result of Test 5- datasheet

Table 6-12 illustrates the summary of tests 1-5 on non-drawing documents. The work package has a lower accuracy compared to the other document types. Bill of Material and Datasheet achieved a higher accuracy.

Table 6-12. Summary of result for title block detection on non-drawing documents

Test no	Number of Documents	Number of document type	Document type	Number of labels	Size of Documents	average precision at 0.5 IoU
1	1000	1	Datasheet	1	Reduce	0.993
2	1000	1	Work Package	1	Reduce	0.855
3	1000	1	Bill of Material	1	Reduce	0.995
4	1000	1	Cable Schedule	1	Reduce	0.962
5	4000	4	Datasheet, Work Package, BOM, Cable Schedule	1	Reduce	0.92

6.3 Test of information extraction model

By developing the title block detection, the location of the title block on each drawing can be found. Then, the title block will be cropped and saved as a new image. In this stage, OCR is applied to the title block to extract the text from the images. Finally, we can classify the documents based on the pre-defined set of keywords.

The keywords are a text file containing all the words that can guide us for classification, such as layout, isometric, map, bill, sheet, etc. The keywords can either be determined by experts or based on historical data.

For the purpose of information extraction, several labels were defined, such as Rev, Document Number, and Document Name. The information extraction model will search the text for pre-defined labels to extract the next token. For example, if the text contains “Rev 2”, the model will find “Rev” and will extract “2” as a revision number. Figure 6-19 illustrates the overview of the proposed information extraction methodology.

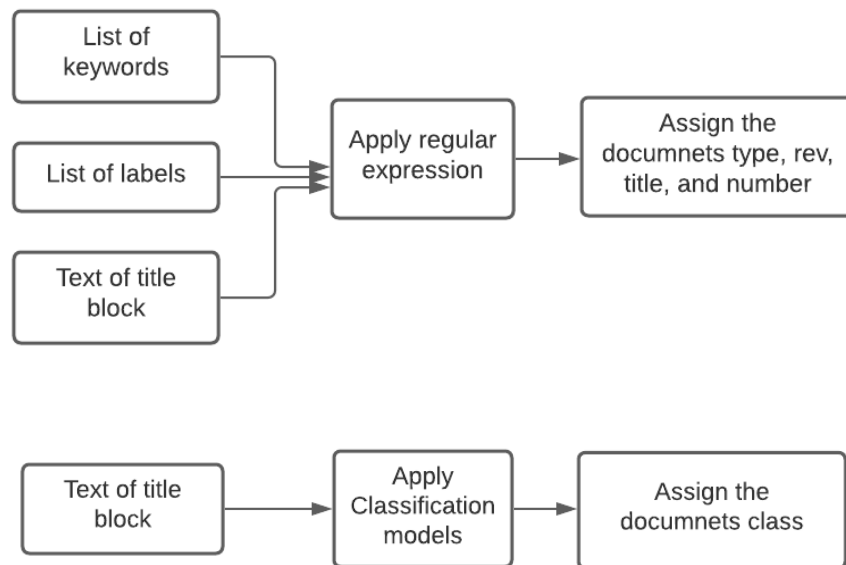


Figure 6-19. Overview of the information extraction model

The test is designed to evaluate the accuracy of the information extraction model. 100 images were selected randomly for the test, including drawing and non-drawing documents. Accuracy on 100 images shows in Table 6-13. The accuracy of document type extraction is 85%, while the accuracy of other information extraction is between 20% to 44%. Figure 6-20 shows the example of a title block that includes all the labels that need to be extracted, such as Rev, Title, Drawing number. And Figure 6-21 shows an example of a title block that does not include any of the labels.

Table 6-13. Result of information extraction

Accuracy	Number of correct answers given by system/ 100
Document type	85%
Title	44%
Rev	20%
Document number	28%
Precision	Number of correct answers given by system/ Number of answers provided by the system
Title	83%
Rev	64%
Document number	62%
recall	Number of correct answers given by system/ Total number of possible correct answers in text
Title	78%
Rev	30%
Document number	31%




 <p>SUNCOR ENERGY</p> <p>THIS DRAWING IS COPYRIGHT AND IS THE PROPERTY OF SUNCOR ENERGY INC. IT MAY NOT BE COPIED, REPRODUCED OR USED IN WHOLE OR IN PART IN ANY WAY WITHOUT WRITTEN PERMISSION OF SUNCOR ENERGY INC. USE OF THIS DRAWING IS PERMITTED ONLY FOR THE SPECIFIC PURPOSE FOR WHICH IT WAS ISSUED BY SUNCOR ENERGY INC. AND IT MUST BE RETURNED IMMEDIATELY UPON REQUEST.</p>			
PC com 42802	TITLE BARRARD PRODUCTS TERMINAL 8" RUL LINE PIPING ISOMETRIC		
	EQUIP. NO. B40	PLANT AREA B40	LINE NO. 40-0-1499-8"-UB1
	DRAWN BY EGB	CHECKED BY NZM	SAP NO. 07-04954
	APPROVED	DRAWING NO. B40-P-IS-01499-01	REV. 0

Figure 6-20. Table of information- A)

	FITTER		INSP. FITTER		INSP. WELDER	
PIPELINE TRAL UTILITIES BLOCK i-13-02-2-P-SS	 <p>Waiward Your Trusted Partner</p>		10030 - 34TH STREET EDMONTON AB , T6B 2Y5 TEL : (780) 469-1258 FAX : (780) 485-3975 WWW.WAIWARD.COM			
			.UMN V-CLM-B	EDITOR SJZ	CHECKER BR	JOB N ^o 19-730
	CATEGORY 9	DRG N ^o 2B-5M				

Received Dec. 09, 2013

Figure 6-21. Table of information- B)

The test result shows that the information extraction model is accurate and useful for the classification of the document. However, the model is unreliable for information extraction such as revision and document number, which is more related to unstructured documents.

Chapter 7 Selected solution

Among all proposed models, three models were selected based on their performance described in chapter 6. This chapter compares the result of three models on a small set of datasets, and then the selected model is applied on a larger-scale data set to determine the model is working for construction companies or not. The first model uses text-based classification, which is consuming crop documents. The OCR engine extracts the text of crop documents, and TF-IDF and Linear SVC were used as document classification algorithms. The second model uses the combination of text-based and image-based classification. Object detection API was applied to detect title block, then OCR engine extracted the text of title block, and TF-IDF and Linear SVC were used in the next step. The third model is based on image-based classification, which uses object detection API and predefined keywords. Figure 7-1 illustrates the comparison between crop, original size, and title block documents were used as an input for method validation. Figure 7-2 the methodology of the three models.

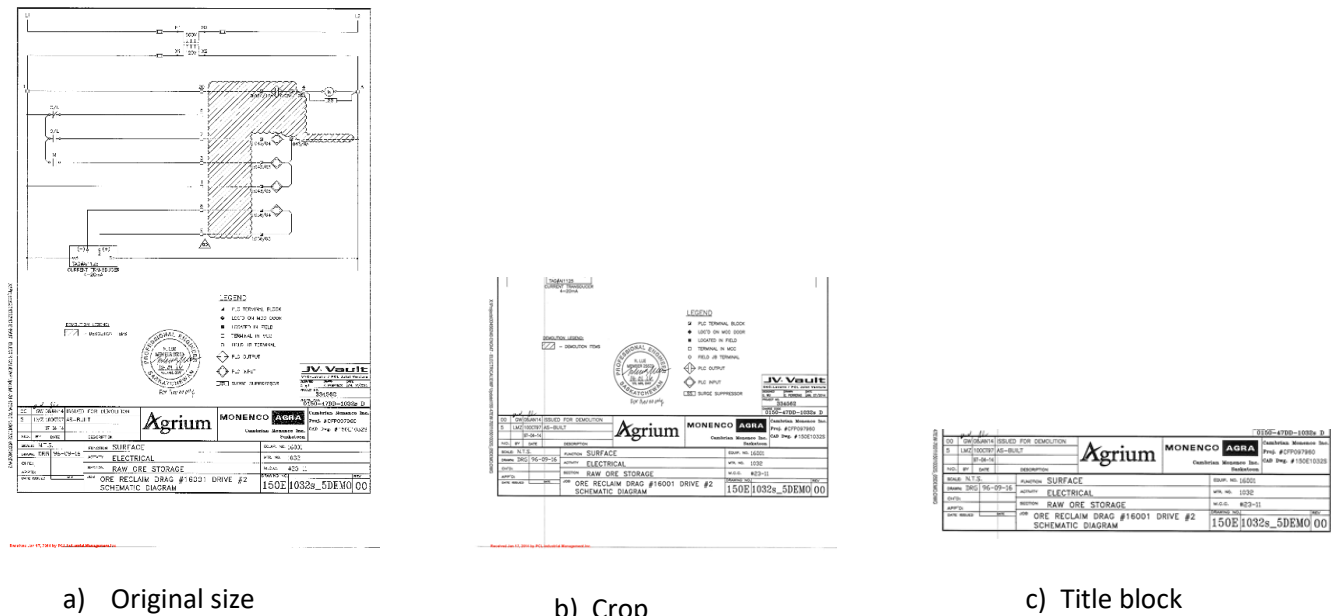


Figure 7-1. Sample of inputs

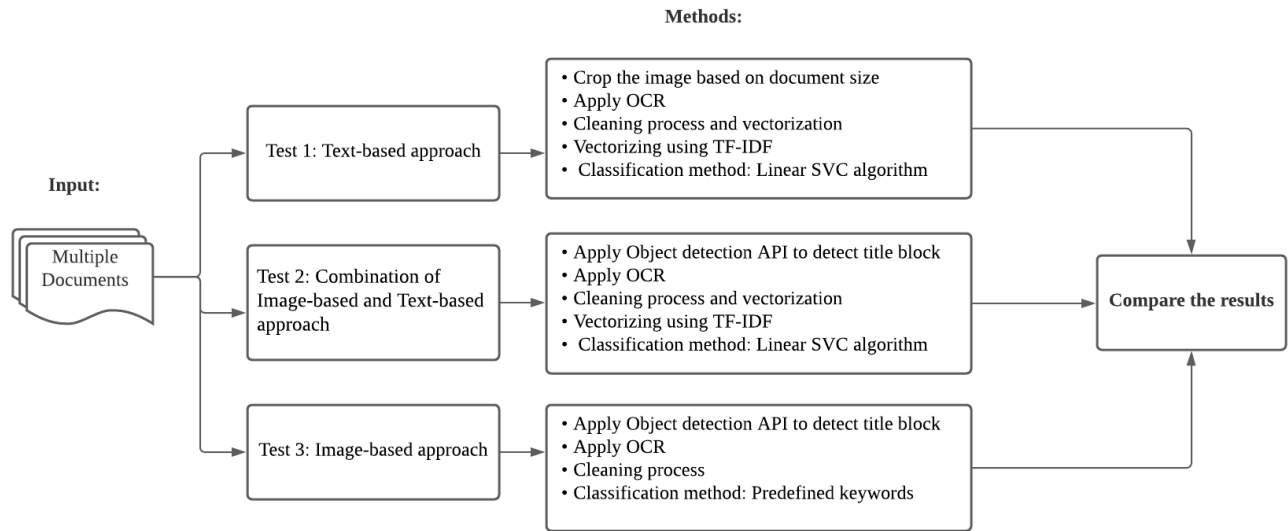


Figure 7-2. Process of three tests

One hundred documents from the new project (never seen before) were randomly selected, which belongs to eight classes: electrical heat tracing (EHT), isometric, layout diagram, loop diagram, piping and instrumentation diagram (P & ID), pipe support, single line diagram (SLD), wiring diagram. the resolution of images should be at least 300 DPI for a better text detection result.

To compare the performance of algorithms, accuracy, recall, and precision were calculated. While accuracy represents the overall performance of the classification model, precision and recall are calculated for each class separately. High precision indicates that an example labelled as positive is indeed positive. For example, the first test 1 shows that among 100 documents, six belong to Pipe support, and false-positive is zero. As a result, the precision of Pipe support is 100%, which means documents that labelled Pipe support are indeed positive. Recall shows how many percentages of actual positive are recognized truly; as a result, the high Recall indicates the class is correctly labelled. Since higher recall means that most documents are labelled correctly, it is more important than the precision metric in this research. For example, the first test 1 shows that among 100 documents, 13 belong to the Loop diagram, and the false negative is zero. As a result, the recall of Loop diagram is 100%, which means all the Loop diagram is recognized, but it may have identified with that some extra documents that are not Loop diagram.

7.1 Comparing the models

The accuracy of the first classification model is 82%; Table 7-1 illustrates the class precision and recall for different document types. Some document types indicate 100% recall, such as isometric and loop diagrams, which means that the classifier could identify all the isometric drawings in the test set, but it may have identified with some additional documents that are not isometric.

Table 7-1. Classification result test 1

Document Type	Recall	Precision
EHT	79	100
Isometric	100	55
Layout	89	80
Loop	100	100
P&ID	86	100
Pipe support	60	100
SLD	65	100
Wiring diagram	100	33

The accuracy of the second classification model is 56%. Table 7-2 illustrates the class precision and recall for different document types. The model could not classify isometric and wiring diagram, therefore recall, and precision is zero.

Table 7-2. Classification result test 2

Document Type	Recall	Precision
EHT	72	86
Isometric	0	0
Layout	72	80
Loop	61	91
P&ID	32	75
Pipe support	66	66
SLD	84	84
Wiring diagram	0	0

The accuracy of the third classification model is 80%. Table 7-3 illustrates the class precision and recall for different document types. Some document types indicate 100% recall, such as wiring and loop diagram, which means their classification was correct on all data sets.

Table 7-3. Classification result test 3

Document Type	Recall	Precision
EHT	88	100
Isometric	92	81
Layout	92	92
Loop	100	91
P&ID	88	88
Pipe support	85	100
SLD	91	100
Wiring diagram	100	92

The first model with an accuracy of 82% and the third model with an accuracy of 80% have a similar performance. Figure 7-3 compares the result of the recall, and Figure 7-4 compares the result of precision among three models.

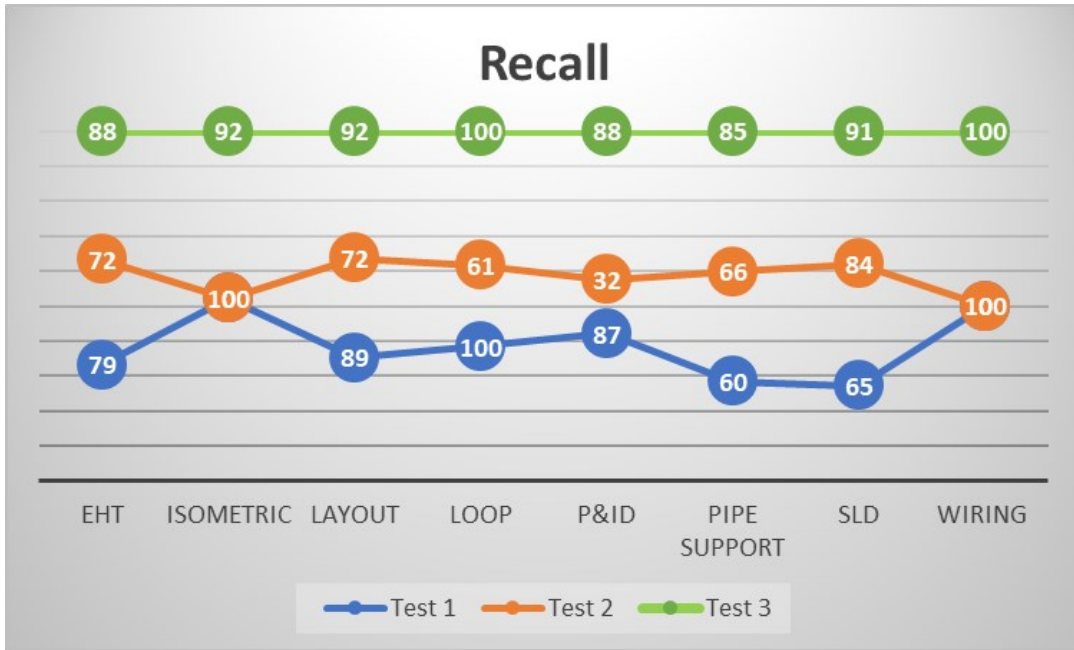


Figure 7-3. Recall results

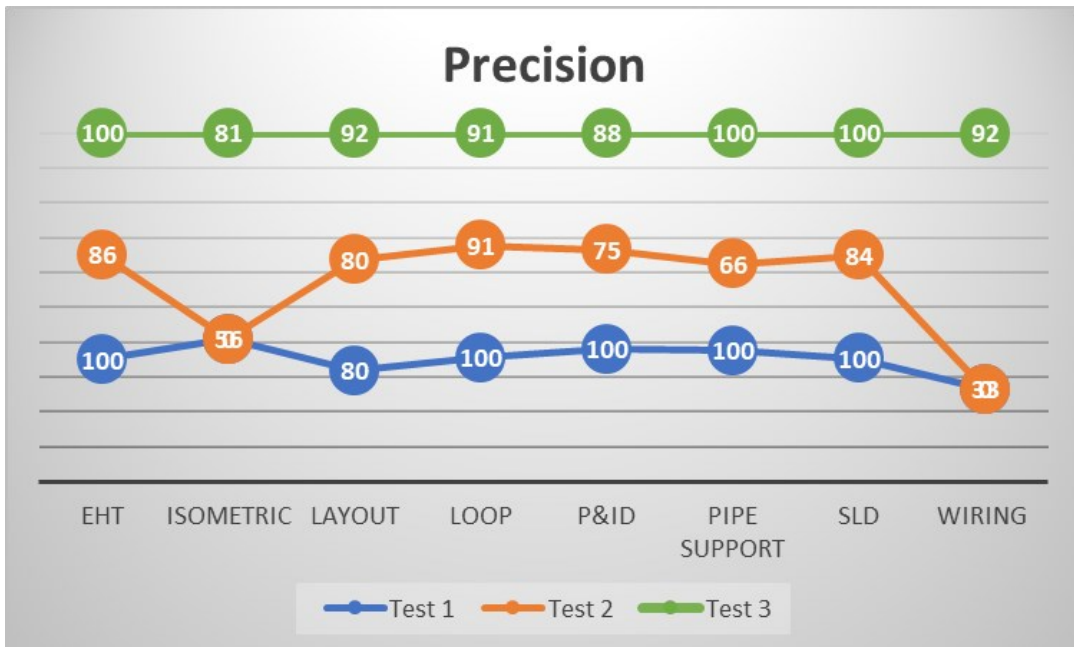


Figure 7-4. Precision results

7.2 Evaluating the selected solution

To determine the reusability of the proposed model, the pre-trained model is applied to a new larger-scale data set. The second data set that was used for evaluating the chosen solution is unique to the model, and the model has never seen this data. The result of the evaluation is used to determine the model is working for construction companies when they have a new set of documents or not. The selected machine learning algorithm is used the OCR engine to extract the text of documents, and TF-IDF and Linear SVC were used as document classification algorithms described in sections 5-1. Figure 7-5 illustrates that in the evaluation section, the model that was trained based on the first data set was applied to the second data set.

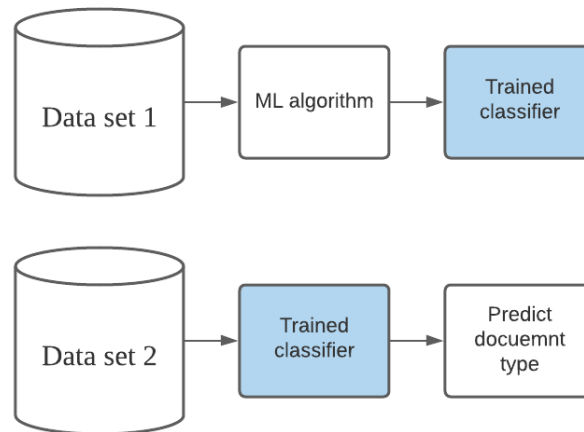


Figure 7-5. Evaluation method

The second dataset was used for the evaluation of the machine learning model, which includes 60,000 construction documents. 32,023 documents belong to eight classes that were used in section 5.1- developing classification based on text. The eight classes are electrical heat tracing (EHT), isometric, layout diagram, loop diagram, piping and instrumentation diagram (P & ID), pipe support, single line diagram (SLD), wiring diagram. Figure 7-6 shows eight document types and the number of documents for each data type. The same as the first data set, the majority of the second data set belong to isometric drawing with 12,404 documents.

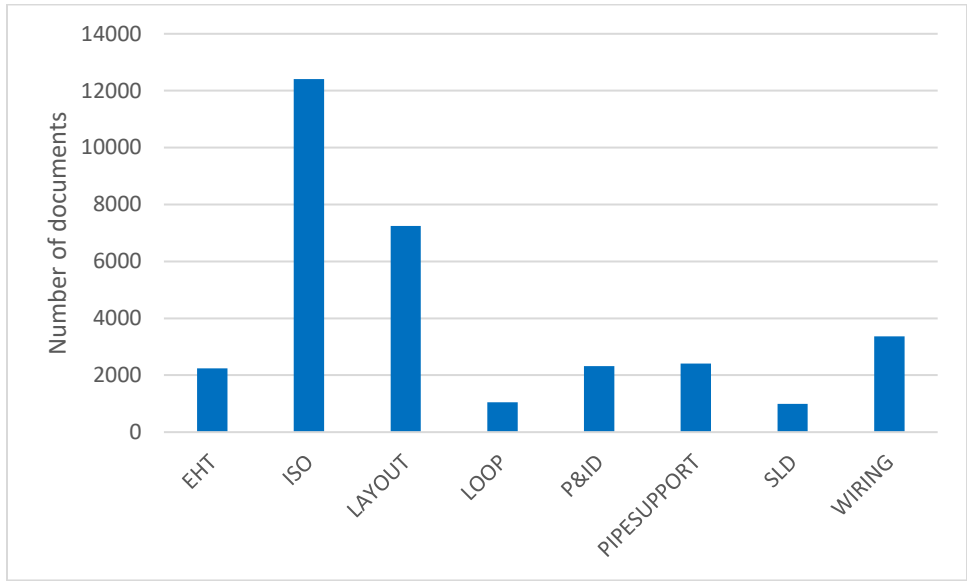


Figure 7-6. Document types in the second dataset

To evaluate the performance of the model, accuracy, recall, and precision were calculated. While accuracy represents the overall performance of the classification model, precision and recall are computed for each class separately.

7.3 Classification results

The accuracy of the classification model on the second data set is 69%; The accuracy of the model is lower than the result of the first data set, which was 97%. However, it approves that the model is working well on a similar construction document template. For instance, the model predicted 11,568 documents correctly among 12,404 isometric drawings. On the other hand, the model is not working well on new templates, and the trained classifier needs to be updated based on new templates. Table 7-4 describes the performance of a classification model by a confusion matrix. In the confusion matrix table, the columns are predicted values, and the rows are actual values.

Table 7-4. Confusion matrix

		Actual value							
		EHT	Isometric	Layout	Loop diagram	P&ID	Pipe support	SLD diagram	Wiring diagram
Predicted	EHT	2181	1	53	0	1	1	0	3
	Isometric	53	11568	2797	61	280	469	41	1346
	Layout	0	15	2123	57	21	8	6	254
	Loop diagram	0	0	9	888	16	0	0	174
	P&ID	1	464	563	5	1915	25	10	345
	Pipe support	1	294	859	1	9	1896	0	3
	SLD diagram	6	51	760	25	62	2	923	534
	Wiring diagram	0	11	87	13	11	6	10	705

Table 7-5 illustrates the class precision and recall for different document types. Wiring and layout document types have a low recall and high precision, which means the model missed many actual wiring and layout. Also, most of the existing wiring and layout are predicted as isometric drawings. The classifier confused between layout-isometric and wiring-isometric. The layout analysis of these two-document types shows that the template of the first and second data sets was different and belongs to other companies. Therefore, we need to add more wiring and layout to improve the accuracy of wiring and layout. Besides these two-document types, the model has an acceptable performance on other document types. EHT, isometric and loop have a high precision and high recall, which means the classifier is working very well on these document types, and almost all of the documents of these types are predicted correctly. SLD has low precision and high recall, which means many of the documents anticipated as SLD are not actual SLD, and they belong to wiring and layout.

Table 7-5. Classification result

Document Type	Precision	Recall
EHT	97	97
Isometric	70	93
Layout	85	29
Loop	82	85
P&ID	58	83
Pipe support	62	79
SLD	39	93
Wiring diagram	84	21

Chapter 8 Conclusion

8.1 Conclusion

In this research, different text-based and image-based classification methods are applied to classify construction drawings documents. The study used a dataset of 160,000 documents that belong to 15 various industrial construction projects. The text-based classification method that involves OCR techniques for text extraction and TF-IDF for text representation, and Linear SVC for document classification achieved the best results on class recall and precision. The implementation outcomes show that despite the poor text content of drawing documents and the lack of structure of this content, the method used still performed very well with an accuracy of 97%, which indicates a potential solution for automating the classification of the document. The evaluation of the model on the new dataset shows that the model is working well when the template is similar to the trained documents, and the model needs to be re-trained when the new template is added to the dataset.

In addition, this research introduced the application of object detection API for construction documents classification and information extraction. The proposed method has three main goals: Title block detection, document classification, and information extraction. This research has presented the first title block detection method on unstructured construction documents. The title block detection approach achieved average precision of 91.7% to 98.8%, depending on the structural complexity of the construction document. The document classification model reached an accuracy of 91.6%. The achieved result approves the potential benefit of using the machine learning approach on construction document classification. However, more information extraction methods need to be tested for increasing the accuracy of information extraction on unstructured documents.

8.2 Contribution

This research addressed the need of construction companies for automated document classification by providing a comprehensive classification model for all types of construction documents with different formats, sizes, and resolutions. While previous studies focused mainly on text-rich documents such as contracts and claims, this research provided appropriate tools for the classification of scanned documents, which have limited text. Based on the literature review, most previous studies rely on well-formed documents where text is readily available in machine-readable format (e.g., PDF, doc, or CAD files); this research automated a text-based approach for information extraction from scanned construction documents. Although it is difficult to access large amounts of data, this study provided 160,000 construction documents as the dataset, including drawing documents such as electrical heat tracing, piping, and layouts, and non-drawing documents for instance bill of materials and work packages. Based on the literature review, other researchers did not provide a large amount of dataset and they provided data set on a few document types. I'm not aware of any suitably big construction document datasets that have previously been used to classify construction documents.

In addition, this research introduced a novel method for title block detection on unstructured construction documents by using object detection API. The result of the automated construction document classification and information extraction model showed that machine learning has the potential benefit of being used in construction document classification and information extraction. The academic contributions can be summarized as follows:

- This research addresses the requirement of construction document classification to support scanned documents as input with different resolutions, sizes, and noises.
- A high accuracy classification model is established for use in the construction domain. The accuracy of the classification model is comparable with the human accuracy level.
- Machine learning tools are employed for the classification of scanned documents that have limited text.

- An automated text-based approach was developed for information extraction from scanned construction documents.
- A domain-specific model is established for both drawing and non-drawing classification.
- A classification model is introduced that includes all types of construction documents.
- An automated approach is introduced for title block detection on unstructured construction documents.

In addition, construction companies can use the proposed model to

- classify drawing and non-drawing documents;
- classify of the document according to the document type;
- identify the location of the title block;
- extract the text of the title block; and
- extract specific information, such as title, name, and revision.

8.3 Limitations and recommendations for future work

The results of the different classifiers indicate that document classification methods that rely on TF-IDF vectors and Linear SVC by achieving 97% accuracy are suitable for classifying scanned construction drawing documents with variable degrees of noise in the image files and limited content of the structured text. The performance of the classification model on direct machine readability of text (i.e., pdf) which does not need OCR steps, should be tested. We expect the performance of the model to be the same or better. Such an approach can automate the document classification task in mega projects that usually include thousands of documents. It helps construction companies to do document classification regardless of the document format.

The result of the tests shows that while the proposed model achieved high accuracy on document classification, the information extraction model was not successful. The information extraction model is accurate and useful for the classification of the document. However, the model is not reliable for

information extraction such as revision and document number, which is related to the nature of unstructured documents. Recommendation to increase the accuracy of information extraction can be summarized as follows:

- Define framework or standard format for documents
- Develop active learning methods
- Combine manual and automated information extraction

References

- (1) Bilal, M.; Oyedele, L. O.; Qadir, J.; Munir, K.; Ajayi, S. O.; Akinade, O. O.; Owolabi, H. A.; Alaka, H. A.; Pasha, M. Big Data in the Construction Industry: A Review of Present Status, Opportunities, and Future Trends. *Advanced engineering informatics* **2016**, *30* (3), 500–521.
- (2) Wolniak, R. Support in ISO 9001:2015. *Zeszyty Naukowe. Organizacja i Zarządzanie / Politechnika Śląska* **2019**, z. 137. <https://doi.org/10.29119/1641-3466.2019.137.16>.
- (3) Lin, H.-T.; Chi, N.-W.; Hsieh, S.-H. A Concept-Based Information Retrieval Approach for Engineering Domain-Specific Technical Documents. *Advanced Engineering Informatics* **2012**, *26* (2), 349–360.
- (4) Smith, C. L. R. Development and Deployment of Document Management Technology into Rover: Executive Summary. PhD Thesis, University of Warwick, 1998.
- (5) Kang, L. S.; Paulson, B. C. Information Classification for Civil Engineering Projects by Uniclass. *Journal of construction engineering and management* **2000**, *126* (2), 158–167.
- (6) Caldas, C. H.; Soibelman, L.; Han, J. Automated Classification of Construction Project Documents. *Journal of Computing in Civil Engineering* **2002**, *16* (4), 234–243.
- (7) Caldas, C. H.; Soibelman, L. Automating Hierarchical Document Classification for Construction Management Information Systems. *Automation in Construction* **2003**, *12* (4), 395–406.
- (8) Mahfouz, T. Unstructured Construction Document Classification Model through Support Vector Machine (SVM). In *Computing in Civil Engineering (2011)*; 2011; pp 126–133.
- (9) Hsu, J. Content-Based Text Mining Technique for Retrieval of CAD Documents. *Automation in construction* **2013**, *31*, 65–74.
- (10) Salama, D. M.; El-Gohary, N. M. Semantic Text Classification for Supporting Automated Compliance Checking in Construction. *Journal of Computing in Civil Engineering* **2016**, *30* (1), 04014106.
- (11) Oevermann, J.; Ziegler, W. Automated Classification of Content Components in Technical Communication. *Computational Intelligence* **2018**, *34* (1), 30–48.
- (12) Al Qady, M.; Kandil, A. Concept Relation Extraction from Construction Documents Using Natural Language Processing. *Journal of Construction Engineering and Management* **2010**, *136* (3), 294–302.
- (13) Al Qady, M.; Kandil, A. Automatic Clustering of Construction Project Documents Based on Textual Similarity. *Automation in construction* **2014**, *42*, 36–49.
- (14) Fard, M. G.; Peña-Mora, F. Application of Visualization Techniques for Construction Progress Monitoring. In *Computing in Civil Engineering (2007)*; 2007; pp 216–223.
- (15) Golparvar-Fard, M.; Bohn, J.; Teizer, J.; Savarese, S.; Peña-Mora, F. Evaluation of Image-Based Modeling and Laser Scanning Accuracy for Emerging Automated Performance Monitoring Techniques. *Automation in construction* **2011**, *20* (8), 1143–1155.

- (16) Wang, C.; Cho, Y. K. Smart Scanning and near Real-Time 3D Surface Modeling of Dynamic Construction Equipment from a Point Cloud. *Automation in Construction* **2015**, *49*, 239–249.
- (17) Rui, Y.; Huang, T. S.; Chang, S.-F. Image Retrieval: Current Techniques, Promising Directions, and Open Issues. *Journal of visual communication and image representation* **1999**, *10* (1), 39–62.
- (18) Rorvig, M. E.; Fitzpatrick, S. J.; Ladoulis, C. T.; Vitthal, S. A New Machine Classification Method Applied to Human Peripheral Blood Leukocytes. *Information processing & management* **1993**, *29* (6), 765–774.
- (19) Flickner, M.; Sawhney, H.; Niblack, W.; Ashley, J.; Huang, Q.; Dom, B.; Gorkani, M.; Hafner, J.; Lee, D.; Petkovic, D. Query by Image and Video Content: The QBIC System. *computer* **1995**, *28* (9), 23–32.
- (20) Bach, J. R.; Fuller, C.; Gupta, A.; Hampapur, A.; Horowitz, B.; Humphrey, R.; Jain, R. C.; Shu, C.-F. Virage Image Search Engine: An Open Framework for Image Management. In *Storage and retrieval for still image and video databases IV*; International Society for Optics and Photonics, 1996; Vol. 2670, pp 76–87.
- (21) Eakins, J.; Graham, M. Content-Based Image Retrieval. **1999**.
- (22) Dinakaran, B.; Annapurna, J.; Kumar, C. A. Interactive Image Retrieval Using Text and Image Content. *Cybern Inf Tech* **2010**, *10*, 20–30.
- (23) Chen, Y.; Wang, J. Z.; Krovetz, R. Content-Based Image Retrieval by Clustering. In *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*; 2003; pp 193–200.
- (24) Le Saux, B.; Boujemaa, N. Unsupervised Robust Clustering for Image Database Categorization. In *Object recognition supported by user interaction for service robots*; IEEE, 2002; Vol. 1, pp 259–262.
- (25) Ke, S.; Zhao, Y.; Li, B.; Wu, Z.; Liu, X. Fast Image Clustering Based on Convolutional Neural Network and Binary K-Means. In *Eighth International Conference on Digital Image Processing (ICDIP 2016)*; International Society for Optics and Photonics, 2016; Vol. 10033, p 100332E.
- (26) Yu, J.; Hong, R.; Wang, M.; You, J. Image Clustering Based on Sparse Patch Alignment Framework. *Pattern Recognition* **2014**, *47* (11), 3512–3519.
- (27) Ozturk, C.; Hancer, E.; Karaboga, D. Improved Clustering Criterion for Image Clustering with Artificial Bee Colony Algorithm. *Pattern Analysis and Applications* **2015**, *18* (3), 587–599.
- (28) Zou, Z.; Shi, Z.; Guo, Y.; Ye, J. Object Detection in 20 Years: A Survey. *arXiv preprint arXiv:1905.05055* **2019**.
- (29) Harzallah, H.; Jurie, F.; Schmid, C. Combining Efficient Object Localization and Image Classification. In *2009 IEEE 12th international conference on computer vision*; IEEE, 2009; pp 237–244.
- (30) Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: A Survey. *International journal of computer vision* **2020**, *128* (2), 261–318.

- (31) Noman, M.; Stankovic, V.; Tawfik, A. Object Detection Techniques: Overview and Performance Comparison. In *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*; IEEE, 2019; pp 1–5.
- (32) Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2014; pp 580–587.
- (33) Girshick, R. Fast R-Cnn. In *Proceedings of the IEEE international conference on computer vision*; 2015; pp 1440–1448.
- (34) Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-Cnn: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in neural information processing systems* **2015**, *28*, 91–99.
- (35) He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-Cnn. In *Proceedings of the IEEE international conference on computer vision*; 2017; pp 2961–2969.
- (36) Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.-Y.; Berg, A. C. Ssd: Single Shot Multibox Detector. In *European conference on computer vision*; Springer, 2016; pp 21–37.
- (37) Redmon, J.; Farhadi, A. Yolov3: An Incremental Improvement. *arXiv preprint arXiv:1804.02767* **2018**.
- (38) Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. In *Proceedings of the IEEE international conference on computer vision*; 2017; pp 2980–2988.
- (39) Abitha A, Lincy K. “A Faster RCNN Based Image Text Detection and Text to Speech Conversion” SSRG International Journal of Electronics and Communication Engineering (SSRG – IJECE) – Volume 5 Issue 5 - May 2018 - (accessed 2021 -07 -08).
- (40) Ch’ng, C.-K.; Chan, C. S.; Liu, C.-L. Total-Text: Toward Orientation Robustness in Scene Text Detection. *International Journal on Document Analysis and Recognition (IJ DAR)* **2020**, *23* (1), 31–52.
- (41) Jiang, Y.; Zhu, X.; Wang, X.; Yang, S.; Li, W.; Wang, H.; Fu, P.; Luo, Z. R2cnn: Rotational Region Cnn for Orientation Robust Scene Text Detection. *arXiv preprint arXiv:1706.09579* **2017**.
- (42) Zhong, Z.; Sun, L.; Huo, Q. An Anchor-Free Region Proposal Network for Faster R-CNN-Based Text Detection Approaches. *International Journal on Document Analysis and Recognition (IJ DAR)* **2019**, *22* (3), 315–327.
- (43) Liu, Z.; Lin, G.; Yang, S.; Feng, J.; Lin, W.; Goh, W. L. Learning Markov Clustering Networks for Scene Text Detection. *arXiv preprint arXiv:1805.08365* **2018**.
- (44) Liao, M.; Zhu, Z.; Shi, B.; Xia, G.; Bai, X. Rotation-Sensitive Regression for Oriented Scene Text Detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2018; pp 5909–5918.
- (45) Nagaoka, Y.; Miyazaki, T.; Sugaya, Y.; Omachi, S. Text Detection by Faster R-CNN with Multiple Region Proposal Networks. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*; IEEE, 2017; Vol. 6, pp 15–20.

- (46) Hao, L.; Gao, L.; Yi, X.; Tang, Z. A Table Detection Method for Pdf Documents Based on Convolutional Neural Networks. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*; IEEE, 2016; pp 287–292.
- (47) Gilani, A.; Qasim, S. R.; Malik, I.; Shafait, F. Table Detection Using Deep Learning. In *2017 14th IAPR international conference on document analysis and recognition (ICDAR)*; IEEE, 2017; Vol. 1, pp 771–776.
- (48) Arif, S.; Shafait, F. Table Detection in Document Images Using Foreground and Background Features. In *2018 Digital Image Computing: Techniques and Applications (DICTA)*; IEEE, 2018; pp 1–8.
- (49) Najman, L.; Gibot, O.; Barbey, M. Automatic Title Block Location in Technical Drawings. In *Proceedings of 4th IAPR International Workshop on Graphics Recognition, Kingston, Ontario (Canada)*; 2001; pp 19–26.
- (50) Najman, L.; Gibot, O.; Berche, S. Indexing Technical Drawings Using Title Block Structure Recognition. In *Proceedings of Sixth International Conference on Document Analysis and Recognition*; IEEE, 2001; pp 587–591.
- (51) Cao, Y.; Li, H.; Liang, Y. Using Engineering Drawing Interpretation for Automatic Detection of Version Information in CADD Engineering Drawing. *Automation in construction* **2005**, *14* (3), 361–367.
- (52) Ondrejcek, M.; Kastner, J.; Kooper, R.; Bajcsy, P. Information Extraction from Scanned Engineering Drawings. *National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign, Image Spatial Data Analysis Group* **2009**.
- (53) Banerjee, P.; Mansoor, S.; Das, S.; Seraogi, B.; Patel, A.; Majumder, H.; Roy, R.; Mukkamala, S.; Chaudhuri, B. B. Automatic Creation of Hyperlinks in AEC Documents by Extracting the Sheet Numbers Using LSTM Model. In *TENCON 2018-2018 IEEE Region 10 Conference*; IEEE, 2018; pp 1667–1672.
- (54) Zhang, J.; Cheng, R.; Wang, K.; Zhao, H. Research on the Text Detection and Extraction from Complex Images. In *2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies*; IEEE, 2013; pp 708–713.
- (55) Berkhahn, V.; Tilleke, S. Merging Neural Networks and Topological Models to Re-Engineer Construction Drawings. *Advances in Engineering Software* **2008**, *39* (10), 812–820.
- (56) Banerjee, P.; Choudhary, S.; Das, S.; Majumdar, H.; Roy, R.; Chaudhuri, B. B. Automatic Hyperlinking of Engineering Drawing Documents. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*; IEEE, 2016; pp 102–107.
- (57) Banerjee, P.; Das, S.; Seraogi, B.; Majumdar, H.; Mukkamala, S.; Roy, R.; Chaudhuri, B. B. Automatic Elevation Datum Detection and Hyperlinking of Architecture, Engineering & Construction Documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*; IEEE, 2017; Vol. 2, pp 37–38.
- (58) Banerjee, P.; Choudhary, S.; Das, S.; Majumder, H.; Mukkamala, S.; Roy, R.; Chaudhuri, B. B. A System for Creating Automatic Navigation among Architectural and Construction Documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*; IEEE, 2017; Vol. 1, pp 677–682.

- (59) Seraogi, B.; Das, S.; Banerjee, P.; Majumdar, H.; Mukkamala, S.; Roy, R.; Chaudhuri, B. B. Automatic Orientation Correction of AEC Drawing Documents. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*; IEEE, 2017; Vol. 2, pp 9–10.
- (60) Gupta, S.; MuNherjee, J.; Bhattacharya, D.; Majumder, H.; Roy, R.; Chaudhuri, B. B. An Efficient Approach for Designing Deep Learning Network on Title Block Extraction for Architecture, Engineering & Construction Documents. *VIENNA* 5.
- (61) Which fields can Procore automatically populate when uploading drawings?
<https://support.procore.com/faq/which-fields-can-procore-automatically-populate-when-uploading-drawings> (accessed 2021 -08 -03).
- (62) Extract Data From PDF: Convert PDF Files Into Structured Data. *Docparser*, 2017.
- (63) PatrickFarley. What is Computer Vision? - Azure Cognitive Services
<https://docs.microsoft.com/en-us/azure/cognitive-services/computer-vision/overview> (accessed 2021 -08 -03).
- (64) Al Qady, M.; Kandil, A. Automatic Classification of Project Documents on the Basis of Text Content. *Journal of Computing in Civil Engineering* **2015**, 29 (3), 04014043.
[https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000338](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000338).
- (65) Webb, G. Naïve Bayes; 2016; pp 1–2. https://doi.org/10.1007/978-1-4899-7502-7_581-1.
- (66) Pal, M. Random Forest Classifier for Remote Sensing Classification. *International Journal of Remote Sensing* **2005**, 26 (1), 217–222. <https://doi.org/10.1080/01431160412331269698>.
- (67) Random Forest Algorithms: A Complete Guide | Built In <https://builtin.com/data-science/random-forest-algorithm> (accessed 2022 -01 -26).
- (68) Jr, D. W. H.; Lemeshow, S.; Sturdivant, R. X. *Applied Logistic Regression*; John Wiley & Sons, 2013.
- (69) Bhavsar, H.; Panchal, M. H. A Review on Support Vector Machine for Data Classification. *Int. J. Adv. Res. Comput. Eng. Technol* **2012**, 185–189.
- (70) Aggarwal, C. C. Neural Networks and Deep Learning. *Springer* **2018**, 10, 978–3.
- (71) Ketkar, N. Convolutional Neural Networks. In *Deep Learning with Python: A Hands-on Introduction*; Ketkar, N., Ed.; Apress: Berkeley, CA, 2017; pp 63–78. https://doi.org/10.1007/978-1-4842-2766-4_5.
- (72) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc., 2012; Vol. 25.
- (73) Reshma Prakash, S.; Nath Singh, P. Object Detection through Region Proposal Based Techniques. *Materials Today: Proceedings* **2021**, 46, 3997–4002. <https://doi.org/10.1016/j.matpr.2021.02.533>.

- (74) Huang, J.; Rathod, V.; Sun, C.; Zhu, M.; Korattikara, A.; Fathi, A.; Fischer, I.; Wojna, Z.; Song, Y.; Guadarrama, S. Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2017; pp 7310–7311.
- (75) Understanding LSTM Networks -- colah's blog <https://colah.github.io/posts/2015-08-Understanding-LSTMs/> (accessed 2022 -01 -25).
- (76) Yin, W.; Kann, K.; Yu, M.; Schütze, H. Comparative Study of CNN and RNN for Natural Language Processing. *arXiv:1702.01923 [cs]* **2017**.
- (77) Sak, H.; Senior, A.; Beaufays, F. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *arXiv:1402.1128 [cs, stat]* **2014**.
- (78) Syeda-Mahmood, T. Extracting Indexing Keywords from Image Structures in Engineering Drawings. In *Proceedings of the Fifth International Conference on Document Analysis and Recognition. ICDAR '99 (Cat. No.PR00318)*; 1999; pp 471–474. <https://doi.org/10.1109/ICDAR.1999.791827>.
- (79) Soibelman, L.; Wu, J.; Caldas, C.; Brilakis, I.; Lin, K.-Y. Management and Analysis of Unstructured Construction Data Types. *Advanced Engineering Informatics* **2008**, *22* (1), 15–27.
- (80) Brilakis, I.; Soibelman, L.; Shinagawa, Y. Material-Based Construction Site Image Retrieval. *Journal of computing in civil engineering* **2005**, *19* (4), 341–355.
- (81) Chi, S.; Caldas, C. H. Automated Object Identification Using Optical Video Cameras on Construction Sites. *Computer-Aided Civil and Infrastructure Engineering* **2011**, *26* (5), 368–380.
- (82) Golparvar-Fard, M.; Heydarian, A.; Niebles, J. C. Vision-Based Action Recognition of Earthmoving Equipment Using Spatio-Temporal Features and Support Vector Machine Classifiers. *Advanced Engineering Informatics* **2013**, *27* (4), 652–663.
- (83) Memarzadeh, M.; Golparvar-Fard, M.; Niebles, J. C. Automated 2D Detection of Construction Equipment and Workers from Site Video Streams Using Histograms of Oriented Gradients and Colors. *Automation in Construction* **2013**, *32*, 24–37.
- (84) Golparvar-Fard, M.; Pena-Mora, F.; Savarese, S. Automated Progress Monitoring Using Unordered Daily Construction Photographs and IFC-Based Building Information Models. *Journal of Computing in Civil Engineering* **2015**, *29* (1), 04014025.
- (85) Fang, Q.; Li, H.; Luo, X.; Ding, L.; Rose, T. M.; An, W.; Yu, Y. A Deep Learning-Based Method for Detecting Non-Certified Work on Construction Sites. *Advanced Engineering Informatics* **2018**, *35*, 56–68.
- (86) Xu, Y.; Wei, S.; Bao, Y.; Li, H. Automatic Seismic Damage Identification of Reinforced Concrete Columns from Images by a Region-Based Deep Convolutional Neural Network. *Structural Control and Health Monitoring* **2019**, *26* (3), e2313.
- (87) Ip, C. Y.; Regli, W. C. Manufacturing Classification of CAD Models Using Curvature and SVMs. In *International Conference on Shape Modeling and Applications 2005 (SMI'05)*; IEEE, 2005; pp 361–365.
- (88) Huang, F. J.; LeCun, Y. Large-Scale Learning with Svm and Convolutional for Generic Object Categorization. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*; IEEE, 2006; Vol. 1, pp 284–291.

- (89) Qin, F. W., Li, L. Y., Gao, S. M., Yang, X. L., & Chen, X. A deep learning approach to the classification of 3D CAD models. *Journal of Zhejiang University SCIENCE C*, 2014, 15(2), 91-106.
- (90) Scheibel, B.; Mangler, J.; Rinderle-Ma, S. Extraction of Dimension Requirements from Engineering Drawings for Supporting Quality Control in Production Processes. *Computers in Industry* **2021**, 129, 103442. <https://doi.org/10.1016/j.compind.2021.103442>.
- (91) Chiang, Y.-Y.; Leyk, S.; Nazari, N. H.; Moghaddam, S.; Tan, T. X. Assessing the Impact of Graphical Quality on Automatic Text Recognition in Digital Maps. *Computers & Geosciences* **2016**, 93, 21–35.
- (92) Oni, O. J.; Asahiah, F. O. Computational Modelling of an Optical Character Recognition System for Yorùbá Printed Text Images. *Scientific African* **2020**, 9, e00415. <https://doi.org/10.1016/j.sciaf.2020.e00415>.
- (93) Sohani, A.; Ullah, R.; Ali, F.; Rao, A.; Messier, R. Optical Character Recognition Engine to Extract Food-Items and Prices from Grocery Receipt Images via Templating and Dictionary-Traversal Technique. *KIET Journal of Computing and Information Sciences* **2019**, 2 (1), 15–15.
- (94) Silva, W. A. J. R.; Shirantha, H. M. K.; Balalla, L. J. M. V. N.; Ranasinghe, R. A. D. V. K.; Kuruwitaarachchi, N.; Kasthurirathna, D. Predicting Diabetes Mellitus Using Machine Learning and Optical Character Recognition. In *2021 6th International Conference for Convergence in Technology (I2CT)*; 2021; pp 1–6. <https://doi.org/10.1109/I2CT51068.2021.9417941>.
- (95) Inglot, J. “Advanced image processing with MATLAB.” (2012).
- (96) CDAR2015 competition on recognition of documents with complex layouts-RDCL2015. In *Document Analysis and Recognition (ICDAR)*, 2015 13th International Conference on (pp. 1151-1155). IEEE.
- (97) Schwenk, K.; Huber, F. Connected Component Labeling Algorithm for Very Complex and High-Resolution Images on an FPGA Platform; Huang, B., López, S., Wu, Z., Nascimento, J. M., Alpatov, B. A., Portell de Mora, J., Eds.; Toulouse, France, 2015; p 964603. <https://doi.org/10.1117/12.2194101>.
- (98) Matas, J.; Chum, O.; Urban, M.; Pajdla, T. Robust Wide Baseline Stereo from Maximally Stable Extremal Regions. 10.
- (99) Adlinge, G.; Kashid, S.; Shinde, T.; Dhotre, V. Text Extraction from Image Using MSER Approach. *03 (05)*, 5.
- (100) Jing, L.-P.; Huang, H.-K.; Shi, H.-B. Improved Feature Selection Approach TFIDF in Text Mining. In *Proceedings. International Conference on Machine Learning and Cybernetics*; IEEE, 2002; Vol. 2, pp 944–946.
- (101) Zhang, J.; Cheng, R.; Wang, K.; Zhao, H. Research on the Text Detection and Extraction from Complex Images. In *2013 Fourth International Conference on Emerging Intelligent Data and Web Technologies*; IEEE, 2013; pp 708–713.
- (102) Feldman, R.; Sanger, J. *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*; Cambridge university press, 2007.

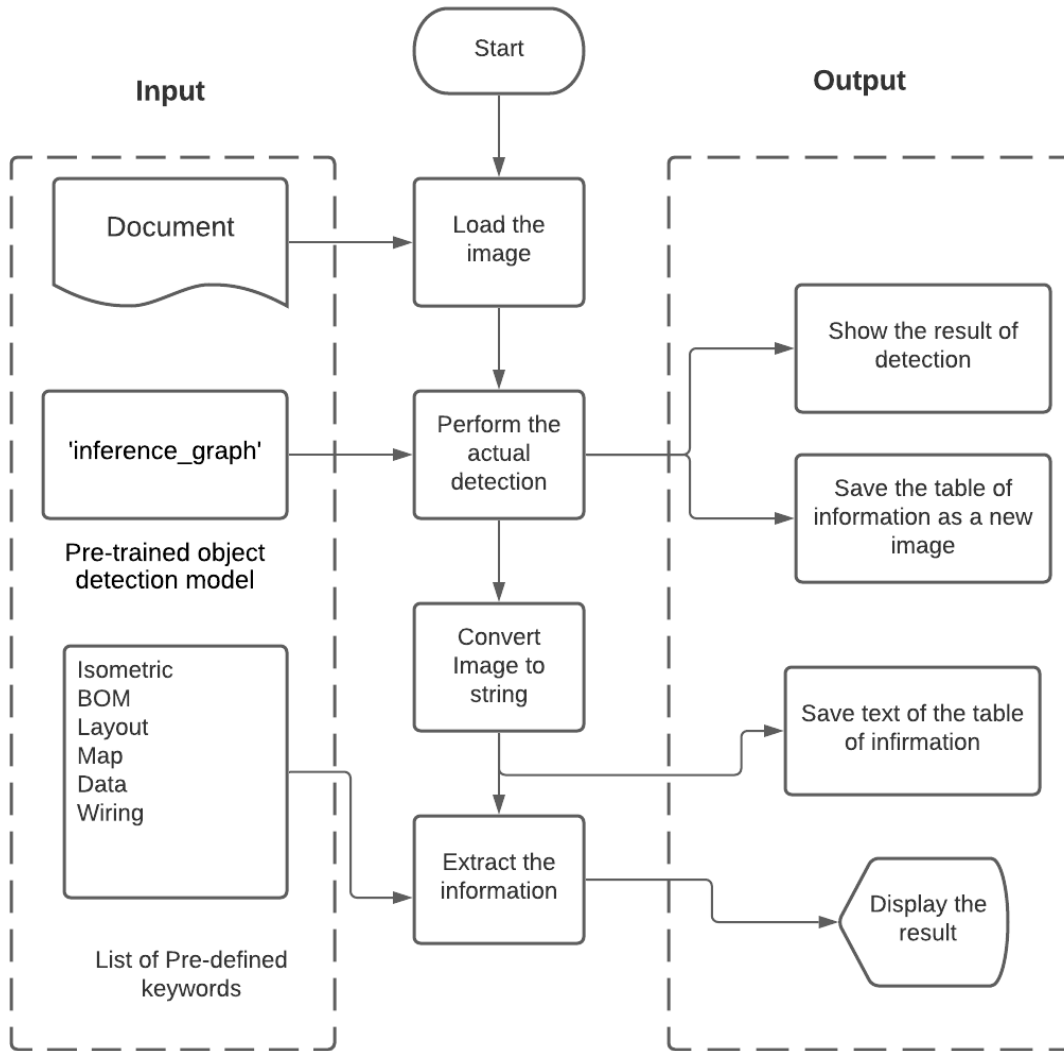
- (103) Bird, S.; Klein, E.; Loper, E. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*; O'Reilly Media, Inc., 2009.
- (104) Vijayarani, S.; Ilamathi, M. J.; Nithya, M. Preprocessing Techniques for Text Mining-an Overview. *International Journal of Computer Science & Communication Networks* **2015**, *5* (1), 7–16.
- (105) Khan, A.; Baharudin, B.; Lee, L. H.; Khan, K. A Review of Machine Learning Algorithms for Text-Documents Classification. *Journal of advances in information technology* **2010**, *1* (1), 4–20.
- (106) Neto, J. L.; Santos, A. D.; Kaestner, C. A.; Alexandre, N.; Santos, D. Document Clustering and Text Summarization. **2000**.
- (107) Gaydhani, A.; Doma, V.; Kendre, S.; Bhagwat, L. Detecting Hate Speech and Offensive Language on Twitter Using Machine Learning: An n-Gram and Tfidf Based Approach. *arXiv preprint arXiv:1809.08651* **2018**.
- (108) Xu, B.; Ye, Y.; Nie, L. An Improved Random Forest Classifier for Image Classification. In *2012 IEEE International Conference on Information and Automation*; IEEE, 2012; pp 795–800.
- (109) Liaw, A.; Wiener, M. Classification and Regression by RandomForest. *R news* **2002**, *2* (3), 18–22.
- (110) Lau, K. W.; Wu, Q. H. Online Training of Support Vector Classifier. *Pattern Recognition* **2003**, *36* (8), 1913–1920.
- (111) Kibriya, A. M.; Frank, E.; Pfahringer, B.; Holmes, G. Multinomial Naive Bayes for Text Categorization Revisited. In *Australasian Joint Conference on Artificial Intelligence*; Springer, 2004; pp 488–499.
- (112) Shimodaira, H. Text Classification Using Naive Bayes. *Learning and Data Note* **2014**, *7*, 1–9.
- (113) Ismiguzel, I. Applying Text Classification using Logistic Regression: A comparison between BoW and Tf-Idf <https://medium.com/analytics-vidhya/applying-text-classification-using-logistic-regression-a-comparison-between-bow-and-tf-idf-1f1ed1b83640> (accessed 2021 -07 -07).
- (114) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. Scikit-Learn: Machine Learning in Python. *the Journal of machine Learning research* **2011**, *12*, 2825–2830.
- (115) M, H.; M.N, S. A Review on Evaluation Metrics for Data Classification Evaluations. *IJDKP* **2015**, *5* (2), 01–11. <https://doi.org/10.5121/ijdkp.2015.5201>.
- (116) Rudin, C. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. *Nature Machine Intelligence* **2019**, *1* (5), 206–215.
- (117) Gunn, S. R. Support Vector Machines for Classification and Regression. *ISIS technical report* **1998**, *14* (1), 5–16.
- (118) Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv:1412.3555 [cs]* **2014**.

- (119) Nowak, J., Taspinar, A., & Scherer, R. (2017, June). LSTM recurrent neural networks for short text and sentiment classification. In *International Conference on Artificial Intelligence and Soft Computing* (pp. 553-562). Springer, Cham.
- (120) Rosander, O.; Ahlstrand, J. *Email Classification with Machine Learning and Word Embeddings for Improved Customer Support*; 2018.
- (121) Tang, D.; Qin, B.; Liu, T. Document Modeling with Gated Recurrent Neural Network for Sentiment Classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Lisbon, Portugal, 2015; pp 1422–1432. <https://doi.org/10.18653/v1/D15-1167>.
- (122) Lu, S.; Lu, Z.; Zhang, Y.-D. Pathological Brain Detection Based on AlexNet and Transfer Learning. *Journal of Computational Science* **2019**, *30*, 41–47. <https://doi.org/10.1016/j.jocs.2018.11.008>.
- (123) Ismail Fawaz, H.; Lucas, B.; Forestier, G.; Pelletier, C.; Schmidt, D. F.; Weber, J.; Webb, G. I.; Idoumghar, L.; Muller, P.-A.; Petitjean, F. InceptionTime: Finding AlexNet for Time Series Classification. *Data Min Knowl Disc* **2020**, *34* (6), 1936–1962. <https://doi.org/10.1007/s10618-020-00710-y>.
- (124) Hosny, K. M.; Kassem, M. A.; Fouad, M. M. Classification of Skin Lesions into Seven Classes Using Transfer Learning with AlexNet. *J Digit Imaging* **2020**, *33* (5), 1325–1334. <https://doi.org/10.1007/s10278-020-00371-9>.
- (125) Almisreb, A. A.; Jamil, N.; Din, N. M. Utilizing AlexNet Deep Transfer Learning for Ear Recognition. In *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP)*; 2018; pp 1–5. <https://doi.org/10.1109/INFRKM.2018.8464769>.
- (126) Zhu, L.; Li, Z.; Li, C.; Wu, J.; Yue, J. High Performance Vegetable Classification from Images Based on AlexNet Deep Learning Model. *International Journal of Agricultural and Biological Engineering* **2018**, *11* (4), 217–223. <https://doi.org/10.25165/ijabe.v11i4.2690>.
- (127) Long Short-Term Memory Networks - MATLAB & Simulink <https://www.mathworks.com/help/deeplearning/ug/long-short-term-memory-networks.html> (accessed 2021 -10 -12).
- (128) darrenl. *Tzatalin/LabelImg*; 2021.
- (129) Mithe, R.; Indalkar, S.; Divekar, N. Optical Character Recognition. *International journal of recent technology and engineering (IJRTE)* **2013**, *2* (1), 72–75.
- (130) Ray, S. *Tesseract-Ocr*; tesseract-ocr, 2019.
- (131) Jones, K. S. A Statistical Interpretation of Term Specificity and Its Application in Retrieval. *Journal of documentation* **1972**.
- (132) Zhang, W.; Yoshida, T.; Tang, X. A Comparative Study of TF* IDF, LSI and Multi-Words for Text Classification. *Expert Systems with Applications* **2011**, *38* (3), 2758–2765.
- (133) Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*; 2016; pp 2818–2826.

(134) Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*; 2019; pp 658–666.

Appendix A- Title block detection

A flowchart of the title block detection model is shown in the figure:



When the model receives a document as input, it will load the document. On the basis of a pre-trained object detection model, the model will detect the title block in the next step. The model will display the result of the object detection model with the bounding box around the table of information. The table of information will be saved as a new image. In the following step, the model will use an OCR engine to extract text from the table of information. Text from the table of information will be saved as well. The model will then extract the information based on the predefined keywords.

Overall, the model has three inputs: A document: 'test.png', List of pre-defined keywords: 'classification.txt' and Pre-trained object detection model: 'inference_graph'. The model has five outputs: title block: '1.test_crop.png', Text of title block: '2.text.txt', Information extraction file: '3.Document information.txt', document class text file: '4.document class.txt' and the original document with a bounding box around the title block: '5.testresult.png'

The original TensorFlow object detection code in Python can find here: [74]

https://github.com/tensorflow/models/tree/master/research/object_detection

Modify code to detect title block:

```
# Import packages
import os
import cv2
import numpy as np
import tensorflow as tf
import sys
```



```

import re

import pytesseract

pytesseract.pytesseract.tesseract_cmd = r'C:\Program Files\Tesseract-OCR\tesseract.exe'

import nltk

from nltk.tokenize import word_tokenize

# This is needed since the notebook is stored in the object_detection folder.

sys.path.append("..")

# Import utilities

from utils import label_map_util

from utils import visualization_utils as vis_util

# Name of the directory containing the object detection module we are using
MODEL_NAME = 'inference_graph'

IMAGE_NAME = 'test.png'

# Grab path to current working directory
CWD_PATH = os.getcwd()

# Path to frozen detection graph .pb file, which contains the model that is used
# for object detection.
PATH_TO_CKPT = os.path.join(CWD_PATH,MODEL_NAME,'frozen_inference_graph.pb')

# Path to label map file
PATH_TO_LABELS = os.path.join(CWD_PATH,'training','labelmap.pbtxt')

# Path to image
PATH_TO_IMAGE = os.path.join(CWD_PATH,IMAGE_NAME)

```

```

# Number of classes the object detector can identify
NUM_CLASSES = 1

# Load the label map.
label_map = label_map_util.load_labelmap(PATH_TO_LABELS)
categories = label_map_util.convert_label_map_to_categories(label_map, max_num_classes=NUM_CLASSES,
use_display_name=True)
category_index = label_map_util.create_category_index(categories)

# Load the Tensorflow model into memory.
detection_graph = tf.Graph()
with detection_graph.as_default():
    od_graph_def = tf.compat.v1.GraphDef()
    with tf.io.gfile.GFile(PATH_TO_CKPT, 'rb') as fid:
        serialized_graph = fid.read()
        od_graph_def.ParseFromString(serialized_graph)
        tf.import_graph_def(od_graph_def, name="")

    sess = tf.compat.v1.Session(graph=detection_graph)

# Define input and output tensors (i.e. data) for the object detection classifier

# Input tensor is the image
image_tensor = detection_graph.get_tensor_by_name('image_tensor:0')

# Output tensors are the detection boxes, scores, and classes
# Each box represents a part of the image where a particular object was detected
detection_boxes = detection_graph.get_tensor_by_name('detection_boxes:0')

# Each score represents level of confidence for each of the objects.
# The score is shown on the result image, together with the class label.
detection_scores = detection_graph.get_tensor_by_name('detection_scores:0')

```

```

detection_classes = detection_graph.get_tensor_by_name('detection_classes:0')

# Number of objects detected
num_detections = detection_graph.get_tensor_by_name('num_detections:0')

# Load image using OpenCV and
# expand image dimensions to have shape: [1, None, None, 3]
# i.e. a single-column array, where each item in the column has the pixel RGB value
image = cv2.imread(PATH_TO_IMAGE)
image_expanded = np.expand_dims(image, axis=0)

# Perform the actual detection by running the model with the image as input
(boxes, scores, classes, num) = sess.run(
    [detection_boxes, detection_scores, detection_classes, num_detections],
    feed_dict={image_tensor: image_expanded})

# save the cropped image
img_h, img_w = image.shape[:2]
ymin, xmin, ymax, xmax = boxes[0][0]
# convert normalized coordinates to image space
ymin, xmin, ymax, xmax = int(ymin * img_h), int(xmin * img_w), int(ymax * img_h), int(xmax * img_w)
image_crop = image[ymin:ymax, xmin:xmax, ...]
CROP_NAME = '1.test_crop.png'
PATH_TO_CROP = os.path.join(CWD_PATH, CROP_NAME)
cv2.imshow(PATH_TO_CROP, image_crop)
cv2.imwrite(PATH_TO_CROP, image_crop)

# Draw the results of the detection (aka 'visulaize the results')
vis_util.visualize_boxes_and_labels_on_image_array(
    image,
    np.squeeze(boxes),
    np.squeeze(classes).astype(np.int32),

```

```

np.squeeze(scores),
category_index,
use_normalized_coordinates=True,
line_thickness=8,
min_score_thresh=0.7)

#Convert image to string
allTexts = str(pyesseract.image_to_string('1.test_crop.png', config=""))
allTexts = allTexts.lower()
print("Title Block Information:",allTexts)
outfile = '2.text.txt'
file1 = open(outfile, "w")
file1.write(allTexts)
file1.close()
allTexts = word_tokenize(allTexts)

# Search for labels
patterns = ["rev", "revision", "name", "number", "title"]
for pattern in patterns:
    print('Looking for "%s" in "%s" = % (pattern, allTexts), end=" ')

    if re.search(pattern, allTexts):
        print("Match was found")
    else:
        print("No match was found")

#Extract the Rev, name, number, title (document information)
allkeys = ["rev", "revision", "name", "number", "title", "no"]
output = {}
for i in allkeys:
    output_internal = []
    for j in range(len(allTexts)):

```

```

if i== allTexts[j]:
    output_internal.append(allTexts[j])
    b = allTexts[j+1]
    print(i +' '+ b)
    outfile = '3.document information.txt'
    file2 = open(outfile, "w")
    file2.write(i +' '+b)
    file2.close()

#Extract the document class
f = open('classification.txt', 'r')
allclassification = f.read().lower().split("\n")
f.close()

#show the match words
print('Document class:')
v = set(allTexts).intersection(allclassification)
print(v)
outfile = '4.document class.txt'
file3 = open(outfile, "w")
z = " ".join(map(str,v))
file3.write(z)
file3.close()

#All the results have been drawn on image. Now display the image.
cv2.namedWindow('Title Block Detector',cv2.WINDOW_FREERATIO)
#image=cv2.resize(image, (1200, 800))
cv2.imshow('Title Block Detector', image)
cv2.imwrite('5.testresult.png', image)
# Press any key to close the image
cv2.waitKey(0)
## Clean up
cv2.destroyAllWindows()

```