Business Information Technology
MSc Thesis

A Comprehensive Study on Feature Detection Algorithms and Fine-Tuning Methods for Efficient Metadata Extraction in Construction Drawings

Pratyush Gupta

Supervisor: Dr. João Luiz Rebelo Moreira | Dr. M.R. Machado

March, 2024

Department of Business Information Technology
Faculty of Electrical Engineering, Mathematics, and Computer Science
University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE.

Acknowledgments

Initially, I extend my deepest gratitude to Dr. João Luiz Rebelo Moreira, Dr. M.R. Machado (Marcos) from the University of Twente, Netherlands, for their unwavering guidance not only throughout the thesis but also during the preparatory Research Topics course. Their constructive feedback and meticulous analysis have been indispensable in the successful culmination of the project. Their assistance in shaping research inquiries tailored to the specific use case has been particularly invaluable.

Furthermore, I am indebted to the Research and Innovation Team at 5ahead, Enschede, and especially to Philip Kwiotek and Leif Conring, for their consistent support throughout the project and for furnishing the essential resources needed for its execution. Leif's oversight in structuring and documenting the project was of immense benefit. I also wish to express my gratitude to another colleague, Faezeh Ali Azadi, for supervising me during the implementation phase of the experiments crucial for addressing the research questions of the thesis.

Additionally, I am profoundly thankful for the unwavering support and encouragement from my brother Yash Gupta, and my family throughout my academic journey, as well as my friends, who provided much-needed diversions during the arduous phases of thesis work. Their support was especially cherished during the challenges posed.

Public Version of Thesis

Important Notice

This is the public version of the thesis submitted by Pratyush Gupta in partial fulfilment of the requirements for the degree of M.Sc. Business Information Technology at the University of Twente. Certain confidential and proprietary information, including but not limited to business models, data, diagrams, and other private information, has been removed from this document. This is in accordance with the policies of the company and the terms of the agreement governing the use of the provided dataset.

This version is intended for public dissemination and may differ from the original version submitted to the university and the company for internal use.

Pratyush Gupta

University of Twente

March, 2024

Contents

List of Figures

List of Tables

List of Abbreviations

CNN : Convolutional Neural Network

CV : Computer Vision

DoG : Difference of Gaussian

ESR : External Sales Representative

GAN : Generative Adversarial Network

GCN : Graph Convolutional Network

GPU : Graphical Processing Unit

HSV : Hue-Saturation-Value

ISR : Internal Sales Representative

JPEG : Joint Photographic Experts Group

LED : Light Emitting Diode

LoG : Laplacian of Gaussian

OCR : Optical Character Reader

ORB : Oriented FAST and Rotated BRIEF

PDF : Portable Document Format

PIL : Python Imaging Library

RLE : Run Length Encoding

RGB : Red-Green-Blue

SLAM : Simultaneous Localization and Mapping

SIFT : Scale Invariant Feature Transform

SURF : Speeded-UP Robust Features

WC : Water Closet

CCOEFF : Correlation Coefficient

CCORR : Cross Correlation

DEI : Detection Effectiveness Index

Abstract

The thesis aims to investigate the potential of leveraging image processing algorithms to extract metadata from construction drawings, with the aim of enhancing efficiency and effectiveness in the industrial processes. The research begins by identifying existing challenges in business process workflows[11], such as lack of automation. Manual extraction of metadata from construction drawings can be time-consuming and error-prone, leading to inefficiencies in the workflow. Through a meticulous review of relevant literature and exploration of image processing methodologies, the study proposes an approach to automate metadata extraction from construction drawings. Central to the research are three key research questions: the feasibility of automating metadata extraction, the identification of optimal detection techniques, and the development of approaches to fine-tune algorithms[40][75] for efficient results. By conducting experiments and evaluations, the study determines that certain image processing algorithms, when optimized through multiple fine-tuning techniques, can effectively extract metadata from construction drawings, even in scenarios with limited data availability. The findings of the research have significant implications for industries reliant on construction drawings, offering insights into cost-effective solutions for enhancing workflow efficiency and customer satisfaction. The optimization efforts resulted in significant achievement in macro-F1 and accuracy scores, with CCORR_NORMED achieving average scores of 0.81(macro-F1) and 0.83(Accuracy) for detection experiments conducted. Also, a proposed index, i.e. Detection Effectiveness Index (DEI) resulted in the score of 0.85 in case of CCOEFF_NORMED. These results highlight the potential of image processing algorithms in addressing real-world challenges, the feasibility of achieving meaningful outcomes with limited data. Finally, contributing to the advancement of knowledge and practices at the intersection of image processing, construction drawings, and industrial processes.

Chapter 1

Introduction

In the rapid evolving landscape of industrial processes, the integration of advanced technologies has emerged as a catalyst for transformative change. Image processing is being one among those technologies and used by big corporations to transform their businesses [63]. Several techniques within image processing can be utilized to optimize its application throughput [23]. However, in the construction business, usage of image data along with computer vision is still a less explored field and needs more involvement of these technologies to reach to its full potential [2]. The purpose of studying the image data can be widely used, but the focus is on how it can be used to make the customer experience better and build a long-lasting relationship with them [41].

The study will focus on the domain of Computer-Aided Design drawings, here after referred as construction drawings due to privacy reasons. It is a transformative field at the intersection of technology, creativity, and engineering precision. Construction drawings represents a revolutionary departure from traditional manual designs methods, introducing a digital realm where ideas are not merely conceptualized but carefully crafted with unprecedented accuracy and efficiency. Navigating through the nuances of construction drawings unveils its pivotal role in shaping the future of design and innovation. Eventually using this innovation and attracting the target market and consumers helps in brand image, popularity and reputation. Leveraging what customer need would help in understanding the market demand and a way to develop technology [66].

Conducted on behalf of small-scale architectural firms seeking to enhance their design processes, this research aims to address a common challenge in the industry: optimizing the reuse of previously designed architectural plans for new clients with similar requirements. These firms, often characterized by their creativity and personalized approach to design, cater to a diverse clientele spanning residential, commercial, and institutional projects. The primary focus of this research lies in harnessing the potential of Computer-Aided Design technology to streamline the adaptation of existing architectural drawings to meet the needs of new clients. By developing a proof of concept applicable to various firms, the goal is to extract relevant visual information from past projects and adapt it seamlessly to new design. This initiative not only aims to enhance efficiency and reduce turnaround time but also to empower small-scale architectural firms to deliver high-quality, customized designs that meet the unique needs of each client. The research aims to decode the complexities of interpreting architectural drawings and articulating their unique features in descriptive terms. This understanding is pivotal for extracting valuable insights from past projects and applying them creatively to new designs across different firms. Ultimately, the overarching objective is to refine the architectural design process industry-wide, ensuring optimal utilization of resources and delivering exceptional outcomes tailored to the vision and requirements of diverse clients.

1.1 Problem Statement and Research Questions

While examining the existing workflow prevalent in small-scale architectural firms, ranging from initial client engagement to finalizing design drafts, several challenges have emerged within the operational framework. Primarily, there is a lack of structured methods for capturing and comprehending client requirements, leading to potential misinterpretations by design teams. This inefficiency often results in significant time wastage, with the generation of new drafts taking an average of two days. Additionally, the iterative process of soliciting client feedback and coordinating revisions with the design team contributes to further delays in project timelines. Moreover, the absence of a mechanism to leverage existing design assets from past projects poses a significant obstacle. Architectural drawings from completed projects are typically stored in databases, identified solely by their drawing numbers. Although each drawing has a unique identifier, it's common for drawings with different names to contain similar design concepts. This redundancy in content could be addressed if metadata were associated with each drawing, facilitating efficient retrieval and reuse of design elements. However, the lack of such metadata hinders efforts to optimize design processes and maximize resource utilization across projects. Businesses offering interior design and furnishing services face challenges in understanding and translating client preferences into actionable design plans [11]. A critical issue arises when construction drawings are available only in PDF format, resulting in the loss of structured data that could be leveraged by technical construction drawing tools. Moreover, the lack of a centralized repository for storing and reusing design assets can result in redundant work and reduced productivity.

To address these challenges, a potential solution involves harnessing the wealth of existing drawings and focusing on creating metadata for these construction drawings. The key emphasis then shifts towards devising a strategy to seamlessly integrate this metadata with the existing architectural framework. An avenue with promising potential to streamline the process is the application of image processing, particularly leveraging detection algorithms such as line detection and object detection. These algorithms within the realm of image processing might hold the promise of providing a viable solution to enhance the overall efficiency and effectiveness of the current workflow. By incorporating these advanced detection techniques, the thesis aims to change the workflow and address the identified issues more proactively. Hence, the key problems boil down as follows:

1. How might advancements in technology be leveraged to improve the extraction of metadata from construction drawings? Addressing this question involves exploring the feasibility of automating the extraction of metadata from construction drawings, particularly those in .pdf format. The challenge lies for the companies in overcoming the absence of two things. First, lack of infrastructure for real-time data capture and storage suitable for model execution and training. Second, readily available metadata for the existing construction drawing visualizations from past designs that were drafted based on certain requirements. Automation could significantly enhance the efficiency of this process, potentially unlocking a valuable resource that has remained untapped due to format constraints.

To achieve that outcome, addressing the below-mentioned research question is necessary. It will provide a clear view to understand and carry out the research.

RQ. Is the extraction of metadata from construction drawings feasible?

This primary research question assesses the feasibility of automating the extraction of metadata from construction drawings, particularly those in .pdf format.

2. What are the algorithms and their threshold parameters that would make the metadata reliable and efficient? In response to this inquiry, the research aims to delve into the selection of algorithms and their associated threshold parameters for ensuring the reliability and efficiency of the extracted metadata. As the focus is on image processing, identifying and fine-tuning algorithms that can accurately interpret visual data from construction drawings becomes crucial. Determining optimal threshold parameters is a key aspect to enhance the precision and relevance of the metadata extraction process.

RQ. What are the optimal detection techniques for extracting details from construction drawings?

The question delves into identifying the most effective object detection techniques suitable for extracting pertinent details from construction drawings. The inquiry involves a careful exploration of various methodologies, considering factors such as accuracy, speed, and adaptability. The objective is to pinpoint techniques that align seamlessly with the distinctive characteristics of construction drawings, ensuring reliable and efficient information extraction.

RQ. What are the approaches to fine-tune the algorithm for efficient results?

Building upon the integration of techniques, the question explores the various approaches to fine-tune the combined algorithm for optimal and efficient results. It involves considering parameters and adjustments that enhance the precision and relevance of the metadata extraction process.

3. What are the parameters that would be used to evaluate the efficiency of the generated metadata? The question addresses the need to establish clear evaluation parameters for assessing the efficiency of the generated metadata. Metrics such as accuracy, completeness, precision, and recall would be considered. By defining and measuring these parameters, the research aims to provide a comprehensive understanding of how well the automated process aligns with the desired outcomes and contributes to the overall optimization of the workflow.

RQ. What are the evaluation parameters to measure the success of the extraction of metadata from construction drawings?

The sub-question emphasizes defining and establishing metrics to systematically evaluate the feasibility and success of the integrated solution. The research seeks to outline measurable criteria for assessing the success and effectiveness of the proposed solution, providing a foundation for a comprehensive evaluation.

Addressing these research questions is integral to solving the challenges outlined in the problem statement. By investigating the feasibility of automating metadata extraction, identifying optimal detection techniques, and exploring integration methods, the research aims to develop a technically sound solution. Fine-tuning the algorithm and establishing evaluation parameters contribute to the precision and effectiveness of the proposed solution. Ultimately, answering these questions will provide insights and innovations that not only address the immediate challenges within the company's workflow but also lay the groundwork for future advancements at the intersection of image processing and industrial processes. Currently, the objective of the research can be formulated as:

<**Design**> the extraction model for the existing construction drawings
<**by**> Implementing the object detection and fine-tuning algorithm
<**that satisfies**> the demand of efficient and relevant metadata
<**to**> serve fast based on the requirements.

The outcome of the research will be the proof of concept to generate metadata that can be used to map the customer queries received via different input methods (Email, phone call, voice message, etc.).

1.2 Research Methodology

The research used an exploratory literature review methodology[58] to explore the field of image recognition, it's history, transformations, methods and applications, and how it has impacted the existing business. This approach allowed for a comprehensive search of the existing literature on the topic, including both theoretical and empirical research, as well as case studies, white papers, and other relevant materials.

The chosen methodology in figure 1.1 aimed to offer a comprehensive and all-encompassing view of the present state of research and practical applications in the domain of image processing. By including a wide range of sources, including grey literature, this study successfully pinpointed essential trends, patterns, and gaps in the existing literature. Additionally, it emphasized specific areas demanding further research attention. Furthermore, the review incorporated innovative technological solutions introduced by various initiatives in the field which are later considered in the implementation phase of the research.

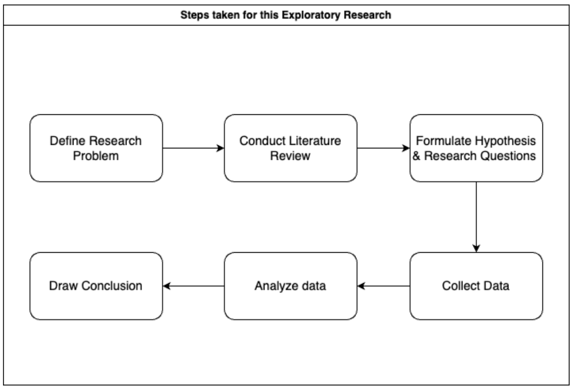

FIGURE 1.1: Exploratory Research Methodology

Also, the research utilized the CRISP-DM (Cross-Industry Standard Process for Data Mining) methodology to structure the project implementation, execution, and evaluation phases as shown in figure 1.2. CRISP-DM provided a systematic framework for addressing the practical aspects of the study, encompassing stages such as data understanding, data preparation, modeling, evaluation, and deployment. Through this methodology, the project team was able to effectively plan and execute the implementation of image recognition solutions, leveraging insights gained from the exploratory literature review. By following a structured approach, including defining project objectives, identifying data sources, preprocessing data, selecting appropriate modeling techniques, and evaluating model performance, the study ensured the successful application of image recognition technologies in real-world business contexts.

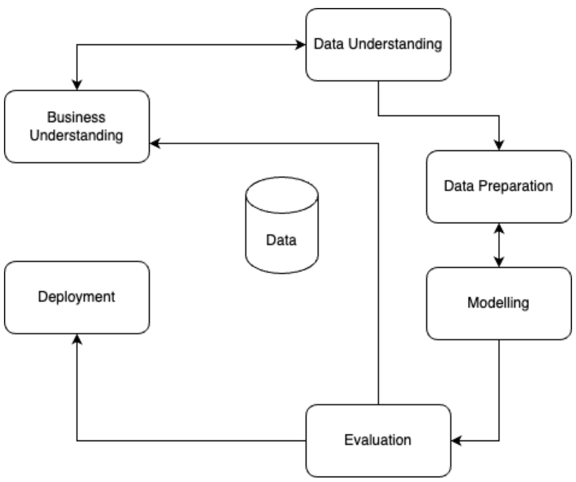

FIGURE 1.2: Crisp-DM Process Diagram[30]

Chapter 2

Literature Review

2.1 Background

The section is organized into four main topics: "construction drawing", "Image Processing Background", "Image Processing Algorithms", and "Fine-tuning Methods". In the first section, a brief history of the introduction of construction drawing is presented and how its involvement changed by time. It also mentions the ongoing researches on construction drawings data extraction. Second, the focus is on understanding the traditional image processing methods and how did they evolve over time. Third, the focus is on discussing four relevant image processing algorithms—SIFT (Scale-Invariant Feature Transform), SURF (Speeded-Up Robust Features), ORB (Oriented FAST and Rotated BRIEF), and Template Matching. Each algorithm is examined in detail, highlighting its applications, technical aspects, and relevance to business operations, particularly in the context of construction drawings. The final section delves into fine-tuning methods for enhancing the performance of the aforementioned algorithms, emphasizing the importance of adaptation in scenarios with limited labelled data. Various innovative approaches, such as transfer learning, resolution handling, colour augmentation, and regularization techniques, are explored.

2.1.1 Construction drawing

Computer-Aided Design has come with a paradigm shift in the field of construction and architecture. Involvement of drawings has not only improved the process in terms of time but also offered unprecedented precision and efficiency in the design process [14]. It marked a transformative shift from the traditional manual designing to digital design platforms. These systems gained prominence in the construction industry during the 1980s with the advent of affordable computing technology [46]. Design software such as AutoCAD emerged as the industry standard, offering architects and engineers powerful tools to create, modify, and analyse complex designs with ease.The integration of this technology in construction processes has yielded numerous benefits. According to Teichoz [59], Technical construction drawing enables architects and designers to visualise concepts in three dimensions, facilitating better communication and collaboration among project stakeholders. Moreover, construction drawings software allows for the generation of detailed drawings and blueprints, streamlining the construction documentation process [46].

Metadata extraction from construction drawings plays a crucial role in enhancing the efficiency and usability of design data. Metadata, which refers to descriptive information about the content and context of a drawing, enables better organization, searchability, and retrieval of design information[46]. Traditionally, extracting metadata from construction drawings has been a manual and labor-intensive process. However, this approach is prone to errors and inconsistencies, leading to challenges in data management and retrieval. By automating metadata extraction[67], construction companies can streamline their design documentation processes, improve data accuracy, and enhance collaboration among project stakeholders. Moreover, automated metadata extraction lays the foundation for advanced data analytics and decision support systems in the construction industry.

Recent advancements in image processing, computer vision, and machine learning have facilitated automated metadata extraction from drawings[10], offering a more efficient and accurate alternative to manual methods. Optical character recognition (OCR) algorithms can analyse images and extract textual information embedded within drawings, such as text labels, annotations, and dimensions [27]. Researches are going on how to systematically use this data from the construction drawings extracted using OCR technology and to build up a semi-automatic geometric digital twinning for existing buildings based on images [36]. In case of Engineering drawings, a conceptual approach is introduced to extract the dimensioning information from the drawings to integrate with the production process [49]. Studies have been conducted on Geometric data on how to dismantle the machine based on the information available and then propose the steps to build it [20]. On metadata POV, researches are under consideration in order to design the semantic model in which information regarding the parts used in the drawings and the adjustment information can be stored so that, similar kind of designs can be engineered again based on the requirements [1]. FloorPlan Computer-Aided-Design is a project which tried to train the model on 10,000 floor drawings by manually annotating them over 11,000 man-hours and proposed a CNN-GCN model to do panoptic symbol spotting task [17].

2.1.2 Traditional Image Processing Methods

In the early stages of image processing during the pre-digital era, particularly in the 1950s, practitioners relied on analogue techniques as their primary tools for basic enhancement and correction of photographs. Compared to digital methods, analogue methods were considered to be inexpensive [8], and the processing time was almost negligible. The prime reason behind it was the expensive resources as computers and the digital processing algorithms to perform those tasks [8]. The era marked the initial attempts to manipulate physical photographs and films through methods such as filtering, masking, and darkroom techniques. Despite its pioneering role, analogue image processing had inherent limitations. The techniques employed were often manual, time-consuming, and lacked the precision and repeatedly achievable with digital methods. The emergence of digital image processing, which followed the analogue era, was driven by the imperative to overcome these limitations and harness the transformative potential of computational approached in handling visual data. The transition to digital methods laid the foundation for more advanced and efficient image processing techniques that continue to shape the landscape of visual data manipulation and analysis later.

During the pivotal period of the 1960 and 1970s, the Image processing field underwent a significant transformation, with the analogue to digital methods. This marked the advent of digital image processing using the point processing techniques, where the focus shifted to pixel-level manipulation. Techniques like contrast stretching [50] and thresholding [47] were introduced, enabling the enhancement of the images by directly manipulating individual pixel values. The transition was more instrumental in providing a more systematic and precise approach to image manipulation. Contrast stretching [50] allowed for the expansion of the dynamic range of pixel intensities, enhancing visual details and improving overall image quality. Thresholding [47] on the other hand, facilitated the segmentation of images by converting them to binary representations based on predefined intensity thresholds. These foundation techniques not only initiated the era of digital image processing but also laid the groundwork for more sophisticated and complex methodologies that emerged in subsequent years. The shift to digital methods signified a paradigm shift in image processing, setting the stage for the development of advanced algorithms and computational approaches that have become integral to modern image analysis and interpretation.

A notable stride was made with the introduction of spatial domain processing techniques during the 1970s and 1980s. The period marked a departure from pixel-level manipulation towards operations in the spatial domain, ushering in the more holistic approach to image enhancement. Spatial domain operations, such as convolution and filtering, emerged as powerful tools for manipulating the spatial coordinates of pixels within an image. Convolution, a fundamental technique, involved overlaying an image with a filter kernel [32] [29], modifying pixel values based on the weighted averages of their neighbouring pixels. Filtering techniques, on the other hand, allowed for the selective modification of pixel values according to the specific spatial criteria. These operations found crucial applications in image smoothing [33], where noise and irregularities were mitigated, fostering a cleaner representation. Additionally, spatial domain processing played a pivotal role in edge detection, highlighting transitions in intensity and identifying boundaries within the image. Furthermore, these techniques were instrumental in basic image enhancement, refining visual features and contributing to the overall clarity of the processed image.

Central to this evolution was the introduction of the Fourier Transform [9], a powerful mathematical tool that facilitated the conversion of image information from the spatial domain to the frequency domain. The Fourier Transform allowed image data to be analysed in terms of its frequency components, revealing patterns and variations that were not readily apparent in the spatial representation. The transformation opened a new avenue for image manipulation and analysis [9]. One key application was in filtering, where operations in the frequency domain enabled the selective modification or removal of specific frequency components. This approach proved particularly effective for tasks such as noise reduction and image enhancement [32], offering a different perspective on image processing challenges. Moreover, frequency domain processing played a crucial role in image compression, where information could be efficiently represented and stored by focusing on essential frequency components. The utilization of the Fourier Transform during this era marked a pivotal advancement in image processing, expanding the toolkit available to practitioners and laying the groundwork for more sophisticated frequency-based techniques in subsequent years.

As technology advanced and digital imaging became more prevalent, the need to extend traditional image processing methods to handle colour images gained prominence. The era from 1990s to 2000s saw the introduction of techniques specifically tailored for colour information, such as separate processing and colour space transformations. Separate channel processing involved the independent manipulation of colour channels, typically red, green, and blue (RGB) [57], allowing for nuanced adjustments to each component. Colour space transformations, on the other hand, provided a means to represent colours in alternative models, such as the Hue, saturation, and Value(HSV) [53] enabling more intuitive and effective manipulation.

Applications of colour image processing during this period were diverse, with notable emphasis on colour correction and segmentation [35]. Colour correction techniques allowed for the enhancement and standardization of colour across images, ensuring a consistent visual experience. In segmentation, the ability to discern and categorise different regions within a colour image became a paramount [35]. These techniques not only addressed the challenges associated with handling colour information but also paved the way for advancements in areas such as computer vision, medical imaging [71], and multimedia applications. The integration of colour specific processing techniques marked a critical phase in image processing, broadening its scope to accommodate the intricacies of colour imagery and contributing to the refinement of various visual data applications.

A distinctive chapter unfolded with the introduction of morphological image processing techniques[21]. Rooted in mathematical morphology, these operations focused on the manipulation of shapes and structures within an image, providing a powerful framework for analysing and enhancing the spatial characteristics of visual data [64]. Morphological operations, such as erosion, dilation, opening, and closing, became fundamental tools in the paradigm [19]. Erosion, for instance, involved the shrinking of object boundaries, while dilation expanded them. Opening, a combination of erosion followed by dilation, was effective in smoothing and breaking narrow connections, while closing, the reverse, filled gaps and eliminated small holes.

The applications of morphological image processing were particularly notable in image segmentation and noise reduction[15]. In image segmentation, morphological operations played a key role in isolating and distinguishing distinct regions or objects within as image based on their shapes and structures. By employing techniques like dilation to connect nearby pixels and erosion to separate interconnected structures, practitioners could achieve refined segmentation results [19]. Additionally, morphological operations proved valuable in mitigating the impact of noise, as the manipulation of shapes and structures allowed for the suppression of unwanted artefacts while preserving essential image features. The era marked the integration of mathematical morphology into mainstream image processing practices, offering a robust set of tools for shape-based analysis and contributing significantly to the fields of computer vision and pattern recognition[42].

During the period spanning from the 2000s to the 2010s, image segmentation emerged as a pivotal technique for delineating and categorizing distinct regions within images. The period witnessed the refinement and widespread adoption of various segmentation methods, tailored to address diverse application challenges [38]. Fundamental techniques like thresholding, involving the classification of pixels into different regions based on greyscale values, were complemented by advanced approaches such as region growing algorithms and clustering methods like K-means [73]. These techniques facilitated object recognition by isolating and identifying individual objects within images [6], aiding scene analysis by parsing complex scenes into meaningful components [6], and enabling critical roles in medical image analysis, particularly in delineating anatomical structures for diagnostic procedures and treatment planning [44]. The refinement and diversification of segmentation algorithms during this period not only contributed to the advancement of computer vision applications but also opened new opportunities for further improvement and research.

In context of different techniques mentioned above, where the emphasis was on granular analysis and manipulation of image structures, the resulting datasets were often large and intricate. The introduction of image compression [68], therefore, became indispensable for efficiently storing and transmitting these extensive datasets. Compression methods were developed to mitigate storage and bandwidth challenges. Lossless compression techniques, exemplified by algorithms like Run-Length Encoding(RLE) [56], aimed at reducing file sizes without compromising any pixel information. However, as the applications of image processing expanded into intricate domains such as morphology and segmentation, the data generated became voluminous and complex. Consequently, lossy compression techniques, typified by standards like JPEG, were introduced [69]. These methods selectively discard redundant or less crucial information, enabling substantial data reduction with acceptable perceptual quality loss. Beyond merely addressing storage concern, compression played a pivotal role in facilitating rapid transmission of image data, a critical requirement in applications such as medical imaging, remote sensing, and multimedia streaming. Technically, the integration of compression techniques into image processing workflows allowed for the optimization of storage resources and facilitated faster transmission of data across networks. This was particularly crucial in scenarios where different manipulation over image data produced large data volumes that were otherwise cumbersome to handle.

The focus shifted towards more advanced methodologies, prominently in the domains of object recognition and computer vision. The period(2010s to present) has witnessed the assimilation of sophisticated techniques such as feature extraction, pattern recognition, and machine learning, which have significantly elevated the precision and efficiency of object-related tasks. Feature extraction involves identifying and capturing relevant information or patterns from images, creating a more condensed representation conducive to subsequent analysis [51]. Technically, the integration of feature extraction, pattern recognition, and machine learning in object related tasks signifies a paradigm shift towards more intelligent and adaptive systems [51]. The ability to understand and interpret visual data in a contextually aware manner has broad implications for fields ranging from robotics to healthcare. As the trajectory of images processing progresses, the fusion of these advanced techniques continues to drive innovations and open new frontiers in the realm of object recognition and computer vision.

Deep learning, particularly through the utilization of deep neural networks like Convolutional Neural Networks (CNNs), has significantly transformed image processing [26]. These networks have revolutionized the field by enabling the automatic learning of hierarchical representations from data. CNNs excel at capturing complex patterns and hierarchies in images by employing convolutional layers. These layers automatically learn features at different scales and abstraction levels, making CNNs particularly adept at tasks such as image classification. In this task, the network can identify objects and assign them to predefined categories with high accuracy [26]. Additionally, CNNs have proven effective in image segmentation tasks, where they delineate and classify different regions within an image.

One notable advantage of deep learning in image processing is its ability to handle large and diverse datasets, facilitating robust model training. The automatic feature extraction capabilities of CNNs reduce the need for manual feature engineering, thereby streamlining the development process. However, this advancement comes with computational demands, and training deep networks often requires significant computational resources [26]. Furthermore, the "black-box" nature of deep learning models can be a limitation, as interpreting the decision-making process may be challenging. Deep learning's applications extend beyond conventional tasks; generative tasks, such as image synthesis and style transfer, have also seen significant advancements. Generative models like Generative Adversarial Networks (GANs) [22] can create realistic images and transform them into various artistic styles.

2.2 Related Work

The exploration of Image processing techniques in the preceding section has laid a comprehensive foundation for understanding the intricacies of advancements and the potential for transformative change. In this section, we continue the narrative by examining the existing object detection techniques that play a pivotal role in the interface between images and efficient workflow optimisation. As discussed, various methodologies have been developed to detect and extract meaningful information from images, contributing to advancements in fields of computer vision. Later, this section includes literature for various fine-tuning methods that can be used to make the algorithms under consideration more robust. In order to make a better solution, the combination of best performed algorithms with the best possible fine-tuning method is the underlying goal of the research.

2.2.1 Object Detection

Scale-Invariant Feature Transform (SIFT) [72] has proven to be a robust and widely utilized algorithm in various technical applications. Its strength lies in identifying distinctive features in an image, regardless of scale and orientation changes, making it suitable for object recognition and image matching. In Architectural point of view as shown in figure 2.1, the template image is gone through smoothing at different levels using Gaussian Scale Space. Difference of Gaussian(DoG) is then calculated between adjacent smoothing samples. Then the Extreme is found among this difference across scales and octaves. Based on extremes, the key points are calculated. For each key point, an orientation is assigned to make the descriptors invariant to image rotation. Once the orientation is assigned, a descriptor is computed for each key point. The region around the key point is divided into smaller sub-regions (usually a grid of cells). Within each sub-region, gradient information is then collected. This gradient information typically includes the magnitude and orientation of gradients. With the strength and direction of these subregions, a concentrated descriptor vector is calculated for each key point. These interest points are compared to the target image at different orientation and scaling, which let the algorithm decide if the template is part of the image.

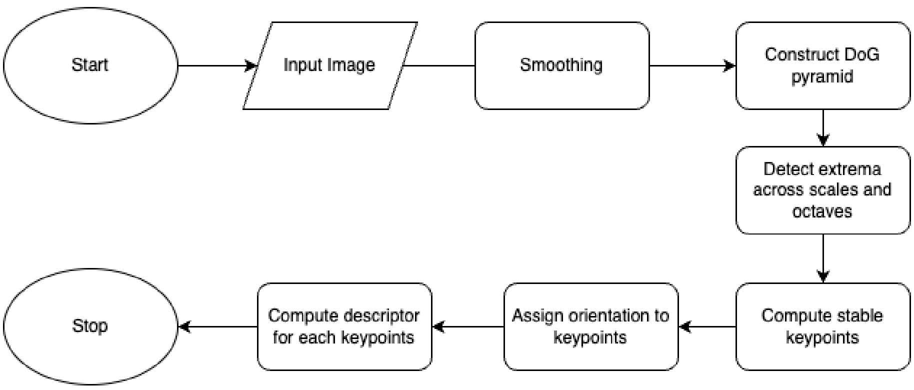

FIGURE 2.1: Flowchart of SIFT Algorithm [31]

SIFT finds applications in recognizing and aligning similar patterns within drawings [62] and videos, aiding in the efficient retrieval of relevant design information. On the business front, SIFT's adaptability extends to tasks like image stitching for panoramic views and video tracking, contributing to enhanced visualizations and simulations. It has also proven a good approach where the images are low in contrast and needs to be compared [55]. In the paper [55], the focus is on adapting the gamma factor for contrast threshold and rescaling factor values for feature matching, in order to get the adequate number of key points that occur in low-contrast and noiseless images. This would be helpful as the current sample data under the consideration is majorly on the white base making it less noisy, and also it is thin black lines along with some coloured templates which makes it low in contrast as well. Its versatility positions SIFT as a valuable asset in automating the extraction of key features drawings, reducing manual efforts and improving overall workflow efficiency.

Speeded-Up Robust Features (SURF) represents an evolution of SIFT [62], designed to enhance computational efficiency while maintaining robustness in detecting image features [43]. In technical domains, SURF finds applications in real-time object recognition, image stitching, and motion tracking. Unlike in SIFT, LoG/DoG filters are used, SURF uses box filters are used to approximate the Hessian. One more major difference is, In SIFT, to get the multiple version of the template the image is scaled, Whereas in SURF the filter is scaled in order to get the multiple version of the template and hence, making it different [5]. To handle the different orientation of images, SURF uses Haar wavelet response in x and y direction instead of using inverse tan in SIFT, and to find the main direction it uses the sliding radial window method to have a more accurate direction [5].

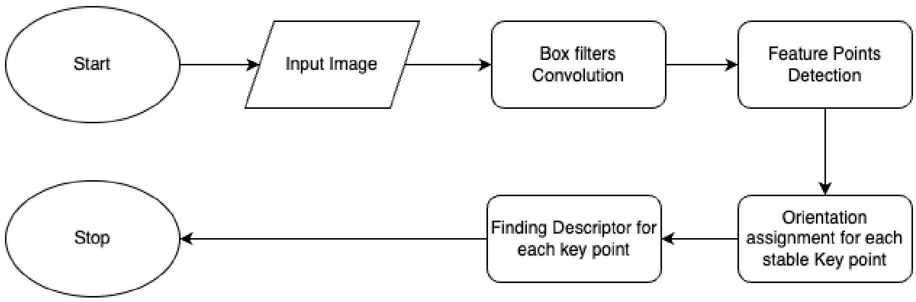

FIGURE 2.2: Flowchart of SURF Algorithm

In the realm of construction business, SURF's speed makes it particularly useful for processing large datasets of drawings swiftly. From a business perspective, its rapid feature detection can expedite the identification of design elements, streamlining the generation of drafts and quotes for potential customers [18]. SURF's balance between speed and accuracy positions it as a pragmatic choice for businesses aiming to integrate image processing seamlessly into their operations. In order to handle large dataset in the current case, considering SURF algorithm can have a positive outcome to generate the metadata quickly and efficiently.

Oriented FAST and Rotated BRIEF (ORB) is tailored for efficiency in terms of both computational resources and memory usage [45]. ORB focuses on enhancing image matching application at low power devices without any GPU acceleration, and also to reduce the time complexity for feature based object detection on standard PCs. Technically, the algorithms first select a pixel and, based on the provided threshold value, it makes a circle of 16 pixels surrounding the main pixel as shown in figure 2.3. These pixels should be above or below the threshold value distance. To make the algorithm faster in detection it first compares the intensity of pixels 1,5,9,13 and at least 3–4 pixels should satisfy the condition in order to check all other remaining pixel values [45]. In order to make it more robust in detecting the differently oriented version. The template image is converted into binary form by rotating the image by 12 degrees and saving the binary output. All these binary outputs are then compared in the target image to check. If the correlation surpasses a given minimum value, then the template is considered to be detected [28]. In technical applications, ORB is employed for real-time object recognition, simultaneous localization and mapping (SLAM), and robotics [28].

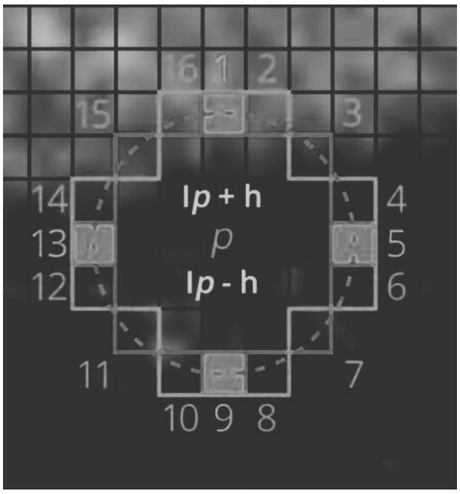

FIGURE 2.3: Image representing keypoint pixel p and the circular area around it[12]

In the business realm, ORB's computational efficiency makes it suitable for handling large databases of construction drawings swiftly. Its applications include rapid pattern recognition, aiding in the automation of initial drafts based on customer requirements. Compared to SIFT, ORB is found to detect more key points. Also, the computation time in ORB is found to be faster than SIFT. SIFT has a good performance in tackling Scale difference issues, whereas ORB is good at handling rotation invariance issues [24]. To deal with contrast issues. This paper [25] suggests an adaptive gamma transform techniques to make the output detection more accurate. ORB's lightweight nature positions it as a practical solution for businesses seeking a balance between computational efficiency [39] and reliable feature extraction in construction-related workflows.

Template matching is a straightforward yet effective technique in which a template image is matched with a larger image to identify occurrences of the template. In technical applications, template matching is used for object detection, image recognition, and localization. As the figure 2.4 shows, the algorithm uses a template image and runs it over the target image like a 2D convolution [37]. It tries to capture the locations which are matched based on the statistical algorithms [36], [54].

In the business context of its application, the method is applied for finding similar patterns within drawings quickly [48]. It facilitates the automation of draft generation by identifying templates from past orders, reducing the time required for manual drafting. Its simplicity and efficiency make it a valuable tool for businesses aiming for a pragmatic approach to object detection within construction drawing

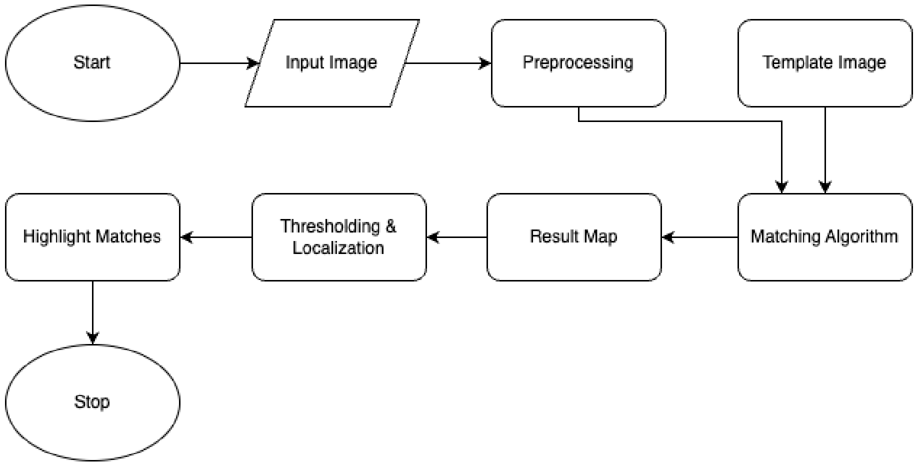

FIGURE 2.4: Template Matching Architecture

images. Also, as discussed above Template matching, which involves sliding the entire image and computing the correlation, is computationally less expensive compared to feature-based methods like SIFT, SURF, ORB.

In comparing these four algorithms, SIFT and SURF exhibit robustness and versatility, with SIFT being slightly more invariant to scaling and rotation. ORB stands out for its computational efficiency, making it suitable for real-time applications [4]. On the other hand, template matching excels in simplicity and speed, but may lack the nuanced feature detection of the other methods. The choice among these algorithms depends on the specific requirements of the computer aided design workflow, balancing factors like accuracy, computational resources, and real-time processing needs. The suitability of each algorithm should be assessed based on the intricacies of the construction drawing images and the desired outcome, allowing businesses to tailor their approach to optimize workflow efficiency.

2.2.2 Fine-tuning Methods

In taking considerations of making the above-mentioned techniques to perform better, fine-tuning algorithms to enhance their performance has become paramount, especially in scenarios where labelled data is scarce. Recent literature reflects a rich tapestry of innovative approaches [52] [74] [34], each contributing to the refinement of algorithms tailored for specific tasks.

The work of Zoph et al. [74] presents Learning Transferable Architectures for Scalable Image Recognition, highlighting the significance of transfer learning in adapting models for scalable image recognition tasks. By incorporating flip-based fine-tuning strategies, this research explores architectures that demonstrate transferable knowledge across diverse datasets and scales. Touvron et al. [60] address the train-test resolution discrepancy in fixing the train-test resolution discrepancy, emphasizing the importance of scaling in image recognition tasks. The research offers insights into strategies for handling resolution variations during fine-tuning, contributing to improved model robustness and generalization.In the context of adversarial attacks, Xie et al. [70] propose Mitigating adversarial effects through randomization. The work introduces thresholding as a randomization technique to enhance model robustness against adversarial perturbations, showcasing its effectiveness in mitigating adversarial effects during fine-tuning.Shorten and Khoshgoftaar [52] present A survey on Image Data Augmentation for Deep Learning, offering a comprehensive overview of data augmentation techniques, including noise injection. This survey emphasizes the role of noise in enhancing model performance, guiding practitioners in selecting suitable data augmentation strategies for their specific tasks.

Cubuk et al. [13], revolutionizes colour augmentation strategies. It provides an innovative approach to automatically learning augmentation policies from data, showcasing the power of colour jittering in improving model generalization and performance during fine-tuning. They also contribute to the field with AutoAugment [13], exploring elastic deformations as part of an augmentation policy. Their approach highlights the efficacy of elastic deformations in capturing diverse image variations, enabling models to adapt effectively to different data distributions during fine-tuning.

Lin et al. [34] present Focal Loss for Dense Object Detection, a seminal work introducing grouping and merging strategies to improve dense object detection. It demonstrates the effectiveness of focal loss in addressing challenges related to object detection, providing insights into techniques that optimize the grouping and merging of visual features during fine-tuning. Devries and Taylor [16] contribute to regularization techniques with Improved Regularization of Convolutional Neural Networks with Cutout. Their work introduces cutout as a regularization method, showcasing its effectiveness in preventing overfitting during training and enhancing model generalization during fine-tuning.

Based on the preceding discussion regarding various techniques employed in recent applications and the methods used to fine-tune these algorithms, the literature review provides a solid foundation for advancing with this knowledge. The algorithms under consideration are slated for testing on construction drawings within the construction department. Factors such as data size, image contrast and intensity, image rotation, speed, robustness, and efficiency must all be taken into account. The resolution of these issues will play a crucial role in determining the course of further implementation aimed at addressing the tentative research questions outlined earlier.

Chapter 3

Metadata Extraction Approach

In tackling the detailed challenges outlined in the problem statement. To find the solution to automate the process for extracting metadata from the drawings with the help of detection algorithms and later to fine-tune those algorithms to improve efficiency. According to the Crisp-DM methodology under consideration, this section strategically breaks down the overarching concern into specific components, aligning with the motivation, research questions and the literature review dealt.

3.1 Data Understanding

The dataset for this research consists of a subset of architectural drawings obtained from various small-scale architectural firms, totalling 93 files. This selection represents a privacy-conscious approach to handling sensitive information, drawn from the firms' extensive collections of architectural drawings. A random sampling technique was employed to select the subset of drawings, ensuring unbiased representation across projects and firms. The chosen drawings are uniformly stored in Portable Document Format (PDF), with the majority being single-pagers, providing concise information without multiple pages.

Each architectural drawing encapsulates various essential components. Firstly, textual information such as project details, dimensions, and specifications are typically located adjacent to the drawing. This section outlines specifics related to the architectural design, including structural elements, materials, and project descriptions. Secondly, the drawings incorporate 3D visualizations, offering comprehensive portrayals from different viewpoints to enhance understanding of the design's spatial configuration. Thirdly, the dataset includes 2D floor plans, containing intricate details essential for metadata extraction. These plans feature markings indicating window dimensions, material specifications, scale indicators, and other relevant information. In cases where architectural elements are intended to be positioned separately, the 2D visualization accommodates such variations, facilitating flexibility in design interpretation. These PDFs are presented in a coloured format, enhancing visual clarity, and are watermarked to protect against unauthorized use. The precise curation of the dataset ensures its integrity, confidentiality, and practicality for research purposes.

3.1.1 Data Distribution

The provided research dataset is organized based on the number of blocks depicted in the PDF visualizations, as outlined in the accompanying table 3.1. The corresponding figure 3.1 visually represents the equal distribution of data across various categories. The focus on block count aligns with the research goal of analysing drawings associated with past orders.

Number of Blocks	Total construction drawings
1	16
2	18
3	19
4	22
5	18
Total	93

TABLE 3.1: Block wise distribution

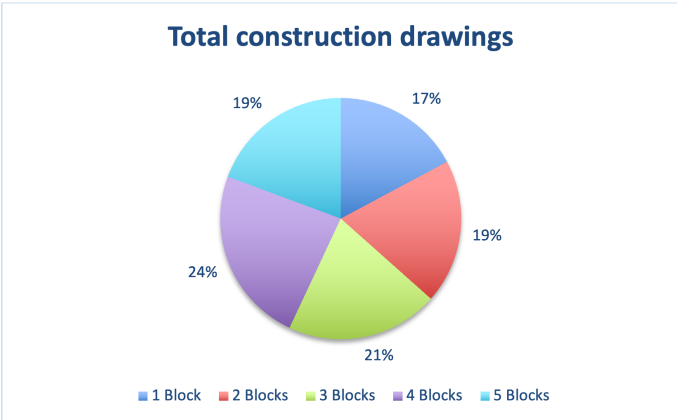

FIGURE 3.1: Data Distribution

Additionally, the figure 3.2 illustrates the dataset's segmentation based on the number of rooms featured in each drawing. While this information is inferred from the provided PDFs rather than explicitly stated, its inclusion is imperative for comprehending and detecting the diverse uses of rooms within the visualizations.

The straightforward data distribution strategy assures a balanced representation of block counts and provides temporal context and insights into room utilization. The clear categorization ensures the dataset's richness, fostering a comprehensive exploration of the complexities inherent in the construction drawings under investigation.

Apart from these drawings, there are 20 more drawings provided to validate the end results. Alike to design-test data, these drawings are also chosen with the

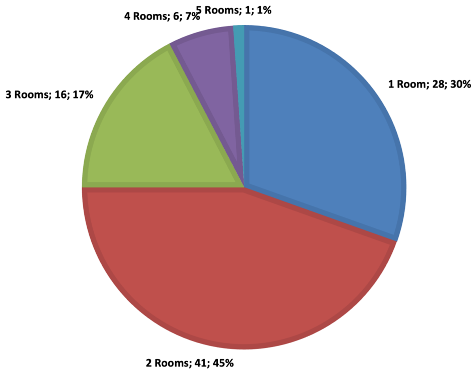

FIGURE 3.2: Room wise distribution

random sampling technique. The selection of these drawings are done on a random basis from the pool of thousands of drawings. The distribution of the drawings based on the number of blocks and rooms are visualised in figure 3.3 & 3.4.

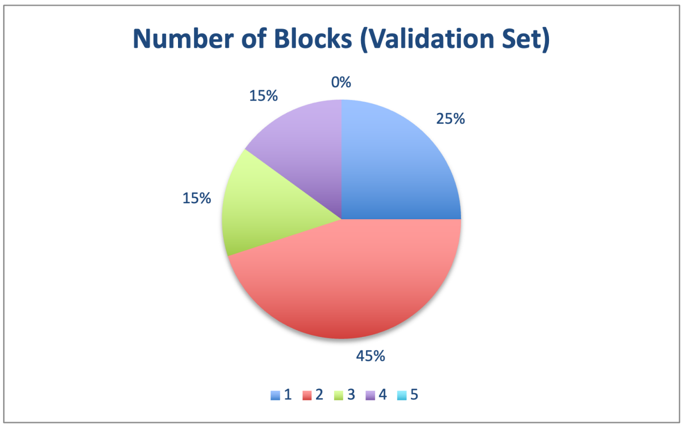

FIGURE 3.3: Validation Set - Blocks Split

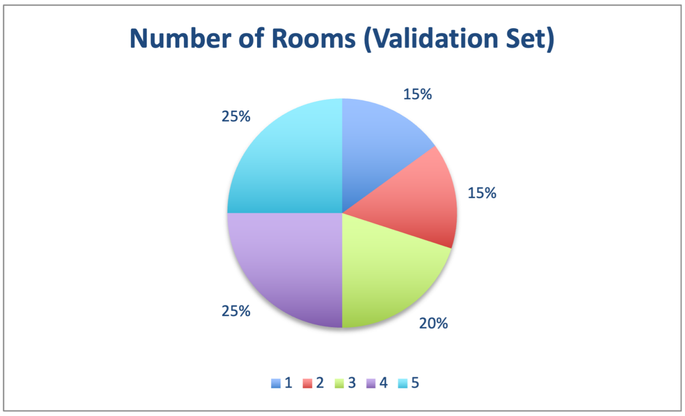

Figure 3.4: Validation Set - Rooms Split

3.1.2 Design-Test Split

In addressing the task of developing a detection algorithm to create metadata from the provided dataset of 93 PDFs, a strategic approach to design-test splitting is paramount. Unlike scenarios where machine learning facilitates autonomous learning, the nature of this algorithm involves human-driven design in which it involves trying out the best algorithm and their fine-tuning approaches in an iterative manner. This would ensure the feasibility by testing over a large proportion of provided data.

Given the distinctive design-centric nature of the algorithm, a counterintuitive approach is employed for the design-test split. Typically, machine learning models learn from a larger portion of the dataset during training; however, in this case, a subset of 35% is dedicated to designing the algorithm, while the remaining 65% is reserved for testing its functionality. The data split involves a distinctive active learning approach[7], emphasizing on selecting the samples with more information to design during the algorithm creation (35% of the dataset) and leaving the ones with less information for evaluation phase (65% of the dataset). This methodology ensures that the algorithm's adaptability and effectiveness are rigorously tested on new data, reflecting real-world scenarios.

The figures 3.5 & 3.6 visually illustrate the unconventional design-test split, reinforcing the deliberate choice made to emphasize algorithm design over autonomous learning. The percentage split of number of rooms and blocks for the design set and the test set is strategically kept similar to get balanced split. The nuanced approach aligns with the research's emphasis on leveraging human insights to enhance the algorithm's capabilities in extracting metadata from construction drawings.

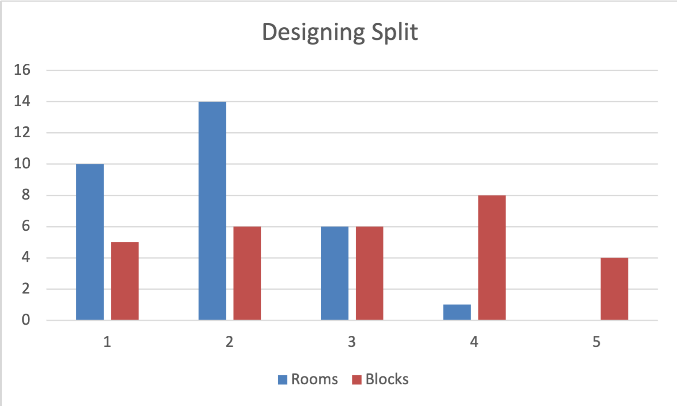

FIGURE 3.5: Designing Split

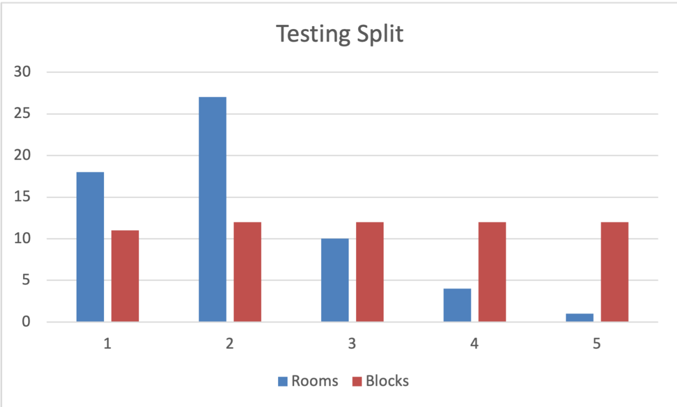

FIGURE 3.6: Testing Split

3.2 Data Preparation

To accurately determine the number of blocks and rooms depicted in the drawings, a comprehensive exploration of the drawings is essential. Initially, a manual examination of the drawings is conducted to identify discernible details that could provide insights into the number of blocks and rooms present. Upon thorough investigation, a list of manually detectable items is compiled.

Upon deeper analysis, it is observed that a specific icon representing distribution boxes, which are electrical units found in each block, can serve as a reliable indicator for detecting the number of blocks within each drawing. As such, the detection of distribution boxes is prioritized for counting the blocks accurately.

Further examination reveals that the drawings predominantly consist of multiple blocks, which are organized to form various categories of rooms. These room categories encompass a range of formations, such as those accommodating urinal and WC facilities, shower rooms, hallways, and more.

To identify and quantify these rooms effectively, the initial step involves the removal of noise present in the drawings. This process aims to isolate the structural boundaries of the rooms, facilitating the subsequent detection and categorization of room types, as well as the determination of the quantity of rooms within each drawing.

By systematically approaching the analysis and processing of the drawings, the goal is to extract meaningful insights regarding the number of blocks and rooms depicted, thus enhancing the efficiency and accuracy of the detection algorithms employed.

3.2.1 Preprocessing Steps

The following steps as shown in figure 3.7 outline the comprehensive approach undertaken to prepare the dataset for algorithmic experimentation.

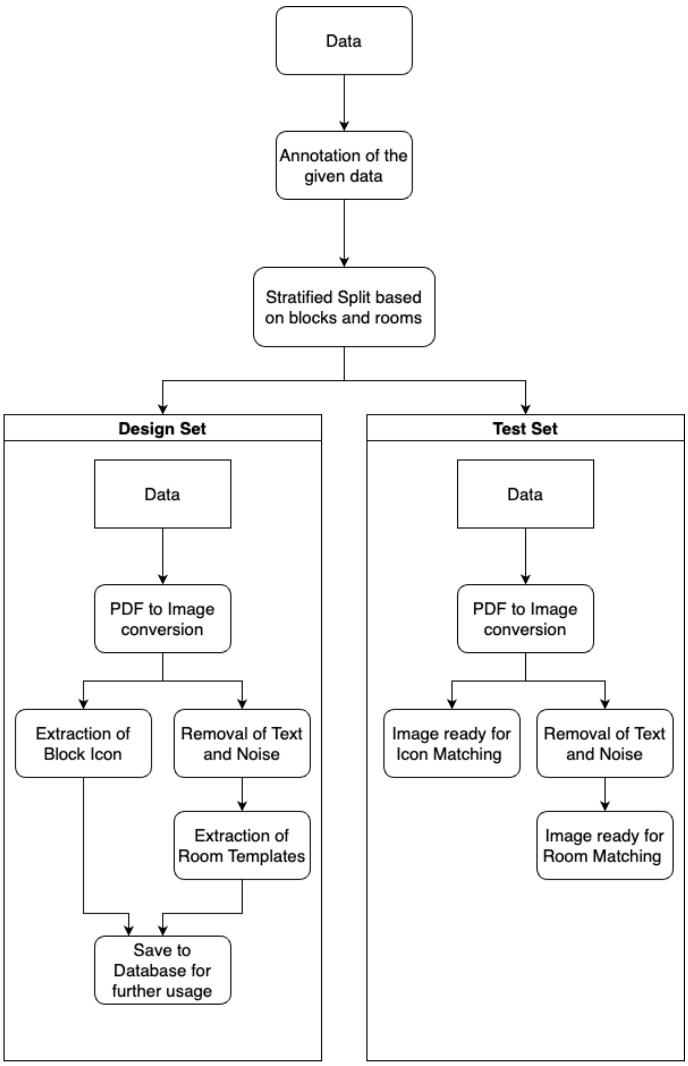

FIGURE 3.7: Preprocessing Steps

1. Manual Data Classification

The initial phase involves the manual classification of the provided construction drawings into distinct categories based on predetermined criteria. Specifically, the drawings are categorized according to the number of blocks and rooms depicted within each visualization. The data set is stratified over number of blocks and rooms, making sure that there are instances of each case in both design and test set.

2. Image Conversion and Processing

Following manual classification, the construction drawings undergo a series of transformations to facilitate their compatibility with image processing algorithms. Initially, the PDF files are read and converted into image format, leveraging established libraries and tools. Subsequently, the images are processed into the Python Imaging Library (PIL) format, a versatile platform for image manipulation and analysis.

3. Icon Extraction

An integral aspect of the preprocessing pipeline involves the extraction of icons essential for experimentation purposes. Leveraging the annotated designing set, specialized cropping functions are employed to isolate and extract icons from the construction drawings. This approach ensures the acquisition of representative icon samples, accounting for variations in scaling and size across the dataset.

4. Room Template Extraction

In parallel with icon extraction, the preprocessing pipeline encompasses the extraction of room templates from the construction drawings. The process entails the removal of extraneous elements, including textual information and embedded icons, to isolate clean room templates characterized by distinct boundaries and internal features.

5. Test Data Processing

Upon completion of icon and room template extraction, the test data undergoes additional processing to prepare it for algorithmic analysis. The test PDF drawings are converted into image format and processed into the PIL[61] format, enabling seamless integration with established image processing algorithms. For icon detection tasks, this transformation suffices to enable template matching and feature extraction.

By articulating the significance and intricacies of the preprocessing pipeline, the experimental setup is positioned for robust and insightful analyses of the construction drawings dataset.

3.3 Model Fine-tuning

In our methodology, the phase dedicated to model fine-tuning holds substantial significance. This stage involves a thorough and iterative process of adjusting various parameters and configurations to enhance the performance of our detection algorithm. The objective is to optimize the algorithm's ability to accurately identify key features within construction drawings. Following the identification of algorithms that meet or surpass our baseline criteria, it becomes imperative to undertake fine-tuning steps. These steps serve as a pivotal mechanism for gaining deeper insights into the behaviour of the algorithms and for further enhancing their performance. After thoroughly understanding the quality and type of drawings that are there in the dataset, the below shortlisted approaches would work in the favour[40][75]. Here, the key parameters considered for fine-tuning are outlined:

Rotation Optimization: construction drawings often contain elements, such as icons and room layouts, that may appear in various orientations. Therefore, incorporating rotation optimization allows the algorithms to effectively detect features regardless of their orientation, thereby enhancing their versatility and adaptability.

Flip: In addition to rotation, construction drawings may also contain instances of flipped icons or room layouts. Considering the possibility of flipped orientations ensures comprehensive detection capabilities for the algorithms, enabling them to accurately identify features irrespective of their orientation.

Threshold Parameter Optimization: Threshold parameters play a crucial role in determining the sensitivity and specificity of detection algorithms. Fine-tuning these parameters involves systematically adjusting them to strike an optimal balance between sensitivity and specificity, thereby ensuring accurate detection of relevant features while minimizing false positives.

Proximity and Grouping Parameters: Fine-tuning parameters related to proximity and grouping is essential for refining how algorithms interpret spatial relationships and group elements within construction drawings. Adjusting these parameters aims to enhance the algorithms' ability to accurately identify and classify features based on their spatial proximity and relationships.

Through a methodical exploration and adjustment of these parameters during the fine-tuning phase, our methodology aims to optimize the algorithms' performance. Ultimately, it will enable the algorithms to achieve superior accuracy and efficiency in detecting and classifying key features within construction drawings.

Chapter 4

Experimentation & Evaluation Setup

4.1 Experiment 1: Baseline

This experiment aims to assess the performance of various algorithms in the context of construction drawing analysis. The objective is to identify the algorithm that exhibits the highest efficacy in feature detection for subsequent metadata extraction. As visualised in figure 4.1 the experiments will be based on the major five feature detection algorithms that came as the output of the literature review namely, SIFT, ORB, TM_CCOEFF_NORMED, TM_SQDIFF_NORMED, TM_CCORR_NORMED. All the algorithms will be considered for the first experiment to find out the better performing algorithm(s). Since the two categories which needs to be detected are the small icons and the room templates. To facilitate the small icons, the experimentation will be performed only on the distribution centre icon that will enable us to detect the number of blocks in the drawing. Whereas for room templates, all the room templates that are extracted from the test set would be considered.

Experiment 1.1 (Block Detection Performance) focus on evaluating how different algorithms perform in detecting the number of blocks within construction drawings. The assessment involves measuring the accuracy and efficiency of each algorithm in accurately identifying and quantifying block elements in diverse drawings.

Experiment 1.2 (Room Detection Performance) delves into the algorithmic performance in detecting the number of rooms within construction drawings. The evaluation considers the algorithms' ability to accurately recognize and quantify room elements, providing insights into their suitability for room-related feature detection.

4.2 Experiment 2: Fine-Tuning Parameter Optimization

The second set of experiments aims to determine the optimal values for fine-tuning parameters that enhance the accuracy and efficiency of feature detection, particularly focused on the detection of various elements and templates within construction

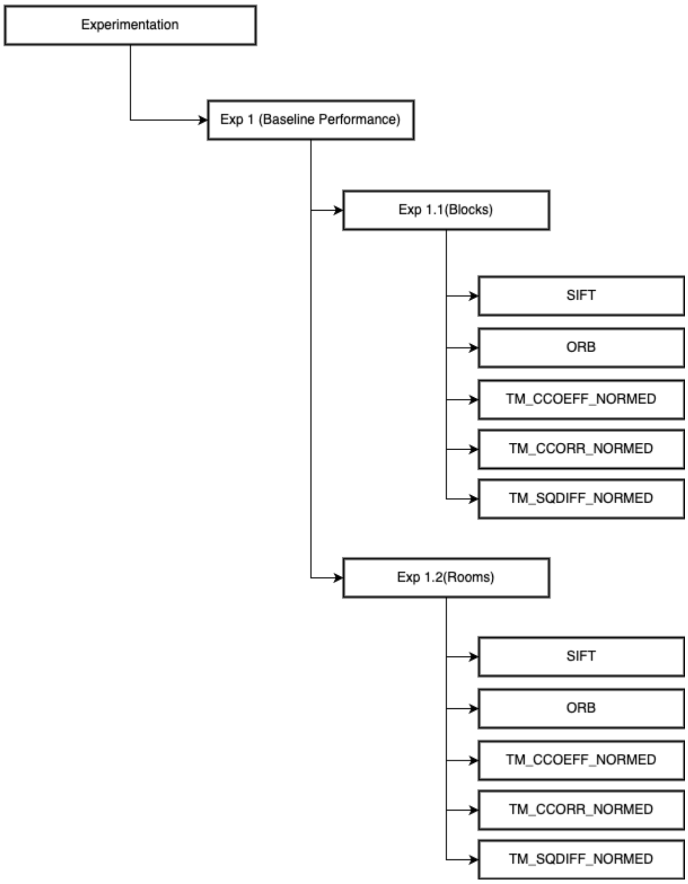

FIGURE 4.1: Experiment 1 Setup

drawings. For this experiment, the best performing algorithms or the ones which has the potential to perform better when fine-tuned from the experiment 1 are considered. The setup of the experiment is visualised in the figure 4.2

Experiment 2.1 (Rotation and Flip Fine-Tuning) investigates the impact of rotation and flip parameters on the accuracy of block detection within construction drawings. By systematically varying parameters, the goal is to identify the values that result in the most accurate and efficient detection of blocks and rooms.

Experiment 2.2 (Thresholding and Grouping Fine-Tuning) shifts to fine-tuning parameters in terms of threshold values and grouping parameters. The experiment aims to uncover the values that optimize the algorithm's performance in recognizing and quantifying block and room elements, contributing to efficient feature extraction.

These experiments collectively contribute to the overarching goal of optimizing the feature detection process within construction drawings. The outcomes will inform the selection of the most suitable algorithm and fine-tuning parameters for automating metadata extraction, thereby addressing the challenges outlined in the problem statement.

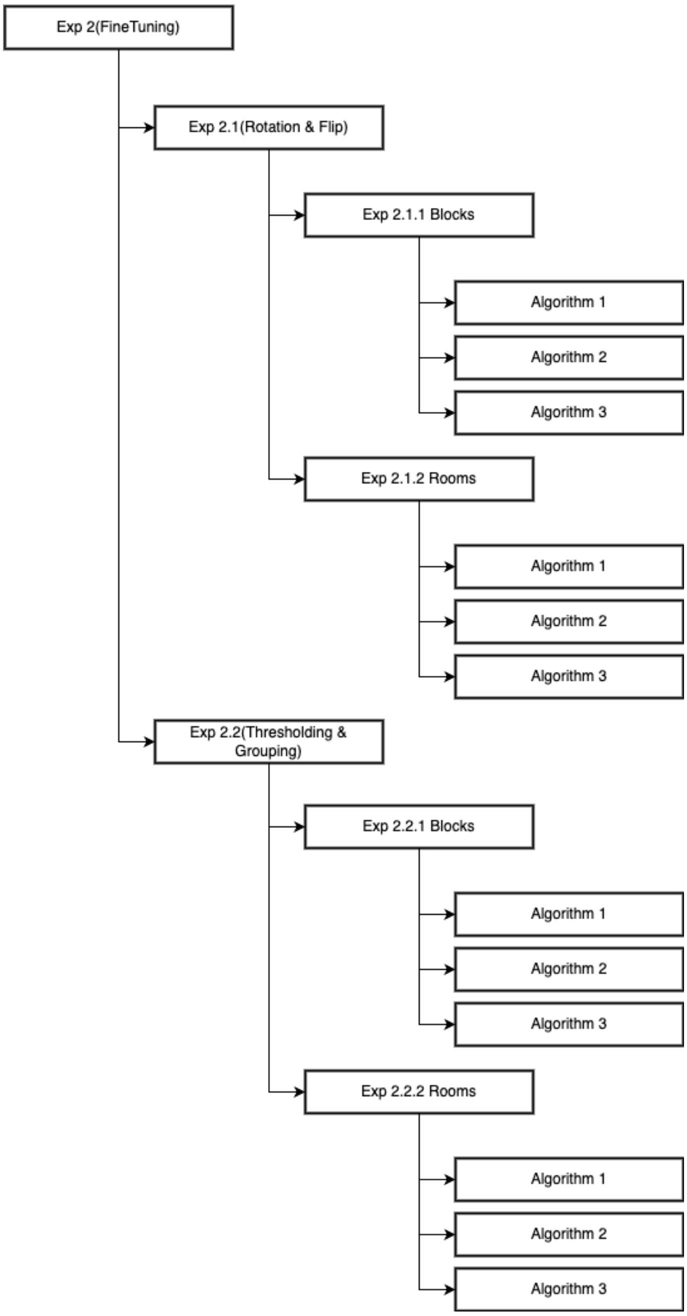

FIGURE 4.2: Experiment 2 Setup

4.3 Evaluation Method

Upon completion of the experiments, a set of evaluation metrics is indispensable to comprehensively understand and analyze the results. Although the experiments encompass the execution of five distinct algorithms aimed at detecting two primary elements – the number of blocks and the number of rooms – a standardized evaluation framework is imperative to facilitate comparative analysis of the outputs.

To provide a comprehensive overview of the results and glean meaningful insights, the evaluation will consider Precision, Recall, Macro Precision, Macro Recall, and Macro F1 metrics[3]. These metrics are defined as follows:

$$\text{Precision} = \frac{\text{All correctly detected items for a target class}}{\text{All detected items for this class}}$$

$$\text{Recall} = \frac{\text{All correctly detected items for a target class}}{\text{All actual items for this class}}$$

$$\text{Macro Precision} = \text{average(Precision for all target classes)}$$

$$\text{Macro Recall} = \text{average(Recall for all target classes)}$$

$$\text{Macro F1} = \frac{2 \times \text{Macro Precision} \times \text{Macro Recall}}{\text{Macro Precision} + \text{Macro Recall}}$$

$$\text{Accuracy} = \frac{\text{All correctly detected instances}}{\text{Total Instances}}$$

The selection of these metrics for assessing the detection of blocks and rooms in construction drawings is well-suited and advantageous within the context of the thesis. Precision and recall metrics offer a detailed evaluation of the algorithm's performance for each class independently, allowing for a nuanced understanding of its capabilities in detecting blocks and rooms separately.Macro-averaged metrics provide a balanced evaluation across all classes, ensuring impartial assessment irrespective of class distribution[3]. This holistic approach considers the performance of detecting both blocks and rooms equally, thereby offering a comprehensive assessment of the algorithm's effectiveness.Furthermore, the F1 score, derived from Precision and Recall, serves to balance the trade-offs between accuracy and completeness, providing a single metric for comprehensive evaluation. Employing standardized evaluation metrics like Precision, Recall, and F1 score facilitates objective comparisons with other methodologies or algorithms, thereby enabling benchmarking and ensuring interpretability and comparability of evaluation results across different studies or implementations.

To get the overall best result for both block and room detection, a metric is proposed considering the macro-F1 scores of the algorithms from both the subtasks of block and room. The metric is referred as "Detection Effectiveness Index(DEI)". It is calculated by considering the harmonic mean of the macro-F1 scores from both the tasks. The formula is represented as

$$\text{DEI} = \frac{2 \times \text{Macro-F1(Block)} \times \text{Macro-F1(Room)}}{\text{Macro-F1(Block)} + \text{Macro-F1(Room)}}$$

In conclusion, the utilization of these standardized evaluation metrics provides a systematic and robust approach to assess the algorithm's performance in detecting blocks and rooms in construction drawings. The approach enables informed decision-making and offers insights for further algorithm refinement, if required.

4.4 Validation Method

Once the experiments are executed, the final step is to validate the results that are produced. The outcome of the experiments would give the tangible parameters like, which algorithms performed well compared to others and what are the values of threshold and grouping range for each of the algorithms which passed the baseline experiment. The validation step aims to execute the algorithms on the validation set which comprises of 20 drawings as discussed in section 3.1.1, with the best parameters found for each algorithm in order to provide the conclusion to results produced in later stages.

Chapter 5

Results and Discussion

This chapter reports the results of the experiments described in experimentation setup. All experiments are executed in a safe environment without using any cloud services to process data and rather used the premise resources in order to meet the guidelines and privacy concern overhead. As mentioned in the previous section, the agenda is to compare different algorithms on two main tasks, namely, block detection and the room detection. The evaluation metrics will be used to analyse the results acquired by the execution of all the experiments and answer the mentioned research questions in section 1.1. For simplicity, the algorithms in our discussion will be addressed as SIFT, ORB, CCOEFF, CCORR, and SQDIFF.

5.1 Experiment 1 — Baseline

5.1.1 Block Detection

The performances of SIFT, ORB, CCOEFF, CCORR, SQDIFF are reported in table 5.1 - 5.5 for block detection. These results are used to compare the possibility of each algorithm to detect the blocks accurately. The mandatory parameters required by each model are kept at 75% accuracy. Which means the results are evaluated by considering how these algorithms perform when the baseline is the best 25% of the results.

TABLE 5.1: **SIFT — Block Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.09	0.36	0.14	11
Class 2	0.20	0.25	0.22	12
Class 3	0.00	0.00	0.00	12
Class 4	0.00	0.00	0.00	14
Class 5	0.00	0.00	0.00	12
Accuracy: 0.11 Macro Avg: 0.06 (Precision), 0.12 (Recall), 0.07 (F1-score) Weighted Avg: 0.06 (Precision), 0.11 (Recall), 0.07 (F1-score)				

Table 5.2: **ORB — Block Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.18	1.00	0.31	11
Class 2	0.00	0.00	0.00	12
Class 3	0.00	0.00	0.00	12
Class 4	0.00	0.00	0.00	14
Class 5	0.00	0.00	0.00	12
Accuracy: 0.18				
Macro Avg: 0.04 (Precision), 0.20 (Recall), 0.06 (F1-score)				
Weighted Avg: 0.03 (Precision), 0.18 (Recall), 0.06 (F1-score)				

Table 5.3: **TM_CCOEFF_NORMED — Block Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.69	1.00	0.81	11
Class 2	0.80	0.67	0.73	12
Class 3	0.91	0.83	0.87	12
Class 4	0.87	0.93	0.90	14
Class 5	1.00	0.75	0.86	12
Accuracy: 0.84				
Macro Avg: 0.85 (Precision), 0.84 (Recall), 0.83 (F1-score)				
Weighted Avg: 0.86 (Precision), 0.84 (Recall), 0.84 (F1-score)				

In table 5.1 it is observed that there are some considerable results from class 1 and class 2. The results suggest that the model was able to detect 1 and 2 block drawings. Given that the block icon is largely similar for all blocks in the drawings, it is highly unlikely that the algorithm cannot accurately detect more than two blocks. Which gives an argument that the model is likely detecting something which looks similar to the block icon but not the icon itself. This is happening due to the internal architectural working of SIFT. When the template is provided to the algorithm to compute stable key points and assign descriptor to it. These descriptors are not enough to compare them with the drawings, which is much larger compared to the template image in terms of dimensions. Although it is scale invariant algorithm, it tries to find the template in the main image as a whole picture. Once, the algorithm creates the descriptors from the main image. Both the patterns from the template image and the main image are then compared. Since these descriptors are calculated based on the extrema across different scales and octaves as discussed in section 2.2.1. Hence, the algorithm is detecting some other pattern which has similar descriptor values but not the template. Because it would have been detecting the template, the results would have spoken for other classes as well.

In table 5.2 it is observed again that the results have got considerable values only for class 1. Since it is understood that the number of blocks that are present

TABLE 5.4: **TM_CCORR_NORMED — Block Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.00	0.00	0.00	11
Class 2	0.00	0.00	0.00	12
Class 3	0.00	0.00	0.00	12
Class 4	0.00	0.00	0.00	14
Class 5	0.20	1.00	0.33	12
Accuracy: 0.20				
Macro Avg: 0.04 (Precision), 0.20 (Recall), 0.07 (F1-score)				
Weighted Avg: 0.04 (Precision), 0.20 (Recall), 0.06 (F1-score)				

TABLE 5.5: **TM_SQDIFF_NORMED — Block Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.69	1.00	0.81	11
Class 2	0.80	0.67	0.73	12
Class 3	0.91	0.83	0.87	12
Class 4	0.93	0.93	0.93	14
Class 5	1.00	0.83	0.91	12
Accuracy: 0.85				
Macro Avg: 0.87 (Precision), 0.85 (Recall), 0.85 (F1-score)				
Weighted Avg: 0.87 (Precision), 0.85 (Recall), 0.85 (F1-score)				

in all the test drawings would range between 1 and 5. The results are also clipped between 1 and 5. The algorithm was not able to detect any block icons from the drawing. The reason turned out again to be the significant difference between the dimensions of the template and the test drawings. As discussed in section 2.2.1 the small template considers each pixel and makes 16 pixel surrounding. Since the block icon is just a rectangle with half white and half black separated diagonally, it is hard to get many descriptors. Also, similar to the SIFT algorithm, the ORB is also trying to compare the whole image with the target image which makes it away from the results what the experiment is trying to achieve as it tries to compare the pattern from the template to the pattern of the whole test image.

In table 5.3 the results are found much better compared to SIFT and ORB. There is a significant detection of blocks for each class. Precision value for Class 5 is turned out to be 1.0 which means that out of drawings which the model detected to have 5 blocks, All the detections are correct. Also, the Recall value of Class 1 is turned out to be 1.0 which means all the drawings that are actually having 1 block in them are accurately detected by the model. The Macro-Precision, Macro-Recall and the Macro-F1 values are more than 0.8 which clearly indicates that the model is able to do the job in our favour.

The results for CCORR are not seemed to be favourable as mentioned in table 5.4. The report claims that the model was not able to detect any significant results for the Class 1-4 and some detection for Class 5. This gives a solid argument that the model is able to detect more than 5 detections in every drawing because the results are clipped between 1 and 5 and there is no activity in Class 1, which clearly states that the model is detecting too many icons than the actual icons. The reason behind this is the accuracy value, As the initial threshold value is set at 75% to find the icons matching better than that value. Therefore, significant results can be observed if this algorithm is considered later in the fine-tuning stage.

In table 5.5 the results are again considerable as there are quite good numbers for each Class. It is visible that the values for each class is almost similar in table 5.3. This shows that the detections are accurate in the majority of the cases. The scores for Macro-precision, Macro-Recall, and Macro-F1 are found to be equal and beyond 0.85 mark which makes it more concrete about moving forward with the algorithm. Also compared to all the 5 models under experimentation, SQDIFF has the maximum accuracy of 85%.

On overall comparison of the baseline performance of all 5 algorithms under consideration, SIFT and ORB does not provide results in our favour. After implementation, it is understood that the template the SIFT and ORB expect needs to have considerable amount of descriptors for detections, and also it requires templates which can have the overall structure of the main image in it. Having only a part of it won't result in better result. COEFF and SQDIFF produced arount 80% of accuracy whereas CCORR only produces about 20% accuracy. Although they belong to the same family, they show drastic variation in the results. This is because that we have considered best 25% detections in all the cases to draw a fair comparison. This provides a valid argument of executing the fine-tuning experiment in later stage to get more accurate results. In the case of CCORR, adjusting the threshold parameter would be necessary to obtain comparable results to CCOEFF and SQDIFF.

5.1.2 Room Detection

The performances of SIFT, ORB, CCOEFF, CCORR, SQDIFF are reported in table 5.6 - 5.10 for Room detection. Before diving into the discussion of individual algorithm performance, here are a few points to consider. Since the focus is to detect the accurate number of rooms, there are two main cases that we need to deal with here. CASE I, there were chances that at certain threshold value different room templates satisfy for the same room structure even though the templates are different. But the total detections are counted based on the number of times the template matches. For, example, if two room templates satisfy the same room in the target drawing, then the total detections would be counted as 2 and not 1 (based on the room identified in the drawing). CASE II, there are chances that we don't have templates for the rooms in the test drawings because the templates for room are extracted based on the drawings considered for designing the drawings, i.e. the design set. In order to gather fair results, the annotation for the rooms is again done based on the templates that we have, and only those rooms are considered for which we have templates from the design set. If we don't do this, the model would be considered faulty because of the data input.

TABLE 5.6: **SIFT — Room Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.50	0.04	0.07	27
Class 2	0.00	0.00	0.00	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.02	1.00	0.03	1
Accuracy: 0.03 Macro Avg: 0.10 (Precision), 0.21 (Recall), 0.02 (F1-score) Weighted Avg: 0.22 (Precision), 0.03 (Recall), 0.03 (F1-score)				

TABLE 5.7: **ORB — Room Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.44	1.00	0.61	27
Class 2	0.00	0.00	0.00	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.00	0.00	0.00	1
Accuracy: 0.44 Macro Avg: 0.09 (Precision), 0.20 (Recall), 0.12 (F1-score) Weighted Avg: 0.20 (Precision), 0.44 (Recall), 0.27 (F1-score)				

In table 5.6 it is visible that almost all values in the classification report are null. The Recall value of Class 5 is 1.0 which means that all the test drawings that have 5 rooms are accurately found by the model. Also, the precision value of class 5 is 0.02, which makes a considerable point, stating that 98% of the drawings which the model predicted to be in Class 5 are misclassified. That means the detections are equal to or more than 5 for the majority of the drawings. The reason behind is the black and white nature of the template and the drawings. Since the template has only black lines with white base, it is hard for the algorithm to detect extrema across different scales and octaves. Although, unlike block icon detection, the size of the room template is comparable to the size of the whole drawing, but since the information is only binary in black and white, it is hard for the algorithm to detect accurately and hence they have the lot of false positive detections.

In table 5.7 it is clear the there is only considerable result for Class 1 and almost null for all the classes. The reason behind this is when the room templates

TABLE 5.8: **TM_CCOEFF_NORMED — Room Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.91	0.37	0.53	27
Class 2	0.08	0.05	0.06	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.06	1.00	0.11	1
Accuracy: 0.20 Macro Avg: 0.21 (Precision), 0.28 (Recall), 0.14 (F1-score) Weighted Avg: 0.43 (Precision), 0.20 (Recall), 0.25 (F1-score)				

TABLE 5.9: **TM_CCORR_NORMED — Room Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.77	0.74	0.75	27
Class 2	0.31	0.20	0.24	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.00	0.00	0.00	1
Accuracy: 0.39 Macro Avg: 0.22 (Precision), 0.19 (Recall), 0.20 (F1-score) Weighted Avg: 0.44 (Precision), 0.39 (Recall), 0.41 (F1-score)				

are considered for extracting key points and descriptors, they are not able to find many. Even though the image is greyscale, the majority of the region is evenly pure white and the only part which represents the boundaries of the room is black. That means majorly there are only two algorithms present in the template, which doesn't provide space for building a good amount of descriptors in the template to compare with the test image. The working of ORB is described in section 2.2.1. Hence, this experiment did not provide satisfactory results to move forward.

CCOEFF has found to perform well in the experiment as shown in table 5.8. As the actual instance of Class 1 is highest and the precision of Class 1 is 0.91 which clears the fact that 91% of the times, the model was right to predict that the drawing has only one drawing. Since the recall of Class 5 is 0.37, this means that the model is able to detect 37% of the actual one block drawings accurately. Also, Precision of 0.06 at Class 5 justifies that the model is detecting more number of rooms for the majority of the drawings compared to the actual number of rooms they have. Since there are considerable results for both the extreme classes, it can be considered for further fine-tuning.

In the table 5.9 we can see significant values for class 1 and 2. There is an ex-

TABLE 5.10: **TM_SQDIFF_NORMED — Room Detection Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.67	0.07	0.13	27
Class 2	0.08	0.05	0.06	20
Class 3	0.17	0.11	0.13	9
Class 4	0.00	0.00	0.00	4
Class 5	0.03	1.00	0.06	1
Accuracy: 0.08 Macro Avg: 0.19 (Precision), 0.25 (Recall), 0.08 (F1-score) Weighted Avg: 0.35 (Precision), 0.08 (Recall), 0.10 (F1-score)				

ception while executing this experiment. The results of this experiment are on the detection of number of rooms when the top 0.01% results are considered. The reason being the internal statistical formula that is used to calculate the cross correlation value for the template and the area under consideration from the target drawing. The recall of Class 2 is 0.2, that means 80% of the actual 2 room drawings are classified wrongly into another classes, which gives a valid point as why the precision values of Class 3, 4, 5 are almost negligible. The majority of the actual instances are from class 1 and class 2 and the precision and recall values of both these classes are turned out to be good, this can be considered for further investigation as it can show better results once fine-tuned.

The results from the SQDIFF algorithm are mentioned in the table 5.10. The small recall values for all the classes 1-4 suggests that the model is not able to predict the actual number of rooms in most of the cases. Also, the precision is in the decreasing nature as we move from Class 1 to Class 5 which gives us the fact that in general model is detecting more number of rooms than the actual number. The reason could be the two cases that are discussed at the start of the section, i.e. more templates might be satisfying the same room structure. To get the output in the range of consideration, detections that fall under best 0.075% are considered. Unlike CCORR, SQDIFF considers the square difference of the pixel values of the template and the considered region of the target drawings. Since the majority of the room templates are having a white base and the drawings under consideration are also having white base, it is hard to have a diversification in correct detections because, almost all the detections would be matching due to lower differences in pixel values.

Based on the 5 algorithms under consideration, and the discussion carried out above for the detecting the number of blocks and rooms, table 5.11 and 5.12 suggests that CCOEFF, CCORR, and SQDIFF would be the best algorithms to consider in order to move with the next Experiment of fine-tuning parameters.

TABLE 5.11: **Block Detection - Baseline Comparison Matrix**

Algorithm	Macro-Precision	Macro-Recall	Macro-F1
SIFT	0.06	0.12	0.07
ORB	0.04	0.20	0.06
TM_CCOEFF_NORMED	**0.85**	**0.84**	**0.83**
TM_CCORR_NORMED	**0.04**	**0.20**	**0.07**
TM_SQDIFF_NORMED	**0.87**	**0.85**	**0.85**

TABLE 5.12: **Room Detection - Baseline Comparison Matrix**

Algorithm	Macro-Precision	Macro-Recall	Macro-F1
SIFT	0.10	0.21	0.02
ORB	0.09	0.20	0.12
TM_CCOEFF_NORMED	**0.21**	**0.28**	**0.14**
TM_CCORR_NORMED	**0.22**	**0.19**	**0.20**
TM_SQDIFF_NORMED	**0.19**	**0.25**	**0.08**

5.2 Experiment 2 — Fine-tuning

This section presents the results about the further fine-tuning experiments conducted over selected algorithms from the Experiment 1. The two subsections represents two steps of fine-tuning. The first section specifies the outcome of the algorithms when rotation and flip detections are introduced, both in block and rooms. The second section specifies further results when the thresholding and grouping parameters are considered for both block and rooms.

5.2.1 Rotation and Flip Fine-Tuning

The results from the CCOEFF, CCORR, and SQDIFF are mentioned in the table 5.13 – 5.15 for block detection and in the table 5.16 – 5.18 for rooms detection. The discussion is drawn based on the evaluation metrics changes compared to the experiment one. All the default parameters are kept intact for respective algorithms. In block detection scenario, best 25% of the results are considered for all the experiments. Whereas in Room detection scenario, it is best 25% for CCOEFF, 1% for CCORR, and 2.5% for SQDIFF consideration. The same values are used in the Experiment 1 so that fair results can be drawn after introduction of Rotation and Flip. The reason of considering different percentages of the best result is the statistical algorithmic difference of each. As the size of the rooms are bigger than the block icon, the pixel level calculations differ in rooms and hence the variation.

Block Detection

TABLE 5.13: **TM_CCOEFF_NORMED — Block Detection Post Rotation and flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.79	1.00	0.88	11
Class 2	1.00	0.83	0.91	12
Class 3	0.92	1.00	0.96	12
Class 4	0.87	0.93	0.90	14
Class 5	1.00	0.75	0.86	12
Accuracy: 0.90 Macro Avg: 0.92 (Precision), 0.90 (Recall), 0.90 (F1-score) Weighted Avg: 0.92 (Precision), 0.90 (Recall), 0.90 (F1-score)				

In CCOEFF, there is an increment of 6% from 84% - 90% of accuracy overall compared to Exp 1. Precision for each class has increased individually and there is considerable increase of from 0.85 to 0.92 in the macro precision value, which means the model is able to improve the detections for each class as it is matching with the true class. The recall values are also increased for individual class and helped macro recall value to go from 0.84 to 0.90. This means that the model can detect the true

Table 5.14: **TM_CCORR_NORMED — Block Detection Post Rotation and Flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.00	0.00	0.00	11
Class 2	0.00	0.00	0.00	12
Class 3	0.00	0.00	0.00	12
Class 4	0.00	0.00	0.00	14
Class 5	0.20	1.00	0.33	12
Accuracy: 0.20				
Macro Avg: 0.04 (Precision), 0.20 (Recall), 0.07 (F1-score)				
Weighted Avg: 0.04 (Precision), 0.20 (Recall), 0.06 (F1-score)				

Table 5.15: **TM_SQDIFF_NORMED — Block Detection Post Rotation and Flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.79	1.00	0.88	11
Class 2	1.00	0.83	0.91	12
Class 3	0.92	1.00	0.96	12
Class 4	0.93	0.93	0.93	14
Class 5	1.00	0.83	0.91	12
Accuracy: 0.92				
Macro Avg: 0.93 (Precision), 0.92 (Recall), 0.92 (F1-score)				
Weighted Avg: 0.93 (Precision), 0.92 (Recall), 0.92 (F1-score)				

detections more accurately for each class. The increase of both macro-precision and macro-recall then pushes the macro-F1 value as well, from 0.83 to 0.90. In CCORR, we cannot see any difference compared to Exp 1 as shown in table 5.14. The reason is similar as stated before, as the 25% of the results under consideration is may be little high because of which there are too many detections compared to the actual blocks in the drawings. The shift is expected once different threshold and grouping values are taken into consideration in the follow-up experiment. In SQDIFF, as shown in table 5.15, it is seen that the values of precision and recall for individual classes have substantially increased from 0.87 to 0.93 for macro-precision, 0.85 to 0.92 for macro-recall, and hence, 0.85 to 0.92 for macro-F1. In a complete view, the algorithms were able to capture the rotated and flipped version of the block icons. The improvement in the results makes sure that there can be instances which represents blocks in different orientations and needs to be considered while designing the proposed algorithm.

Room Detection

In CCOEFF, the results obtained from this experiment exhibit an inversion compared to the anticipated outcome. The recall of class 1 is decreased from 0.37 to

Table 5.16: **TM_CCOEFF_NORMED — Room Detection Post Rotation and flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.88	0.26	0.40	27
Class 2	0.00	0.00	0.00	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.04	1.00	0.08	1
Accuracy: 0.13				
Macro Avg: 0.18 (Precision), 0.25 (Recall), 0.10 (F1-score)				
Weighted Avg: 0.39 (Precision), 0.13 (Recall), 0.18 (F1-score)				

Table 5.17: **TM_CCORR_NORMED — Room Detection Post Rotation and flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.76	0.70	0.73	27
Class 2	0.31	0.20	0.24	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.09	1.00	0.17	1
Accuracy: 0.39				
Macro Avg: 0.23 (Precision), 0.38 (Recall), 0.23 (F1-score)				
Weighted Avg: 0.44 (Precision), 0.39 (Recall), 0.41 (F1-score)				

0.26, that means when the rotation and flip is introduced, then in the case of one room drawings, model started to detect more number of rooms and hence there is a decrease in recall value. But the point is, how can rotation and flip would result in detection less accurately? The answer is, since our results are clipped between 1 and 5, the cases where the rooms were not detected and were counted in Class 1 as the matter of fact that each drawing would always have at least 1 room. Now, as the rotation and flip is introduced, these rooms are able to get detected, thus increasing the respective room count and lowering the value of recall in Class 1. Overall, as we can see that the precision value of Class 5 is only 0.04, which means in the significant amount of the cases the rooms detected by the algorithm is 5 and beyond, and also we can see activity in class 1. Since, the results are visible for both the extreme classes, there is a high chance that the result would improve when other fine-tuning parameters would be considered along with this. In the CCORR scenario as well, we can see the similar shift and the reason is also similar. Also, in the case of SQDIFF, the values for all the classes except class 5 is null. The precision value of 0.02 and the recall value of 1.0 in Class 5 assures that the model is detecting way to many rooms than the actual room. Compared to Exp 1, when Flip and Rotation is introduced now, the model is mapping different flipped version of the template to the same orientation room, because of which compared to Exp 1

TABLE 5.18: **TM_SQDIFF_NORMED — Room Detection Post Rotation and flip Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.00	0.00	0.00	27
Class 2	0.00	0.00	0.00	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.02	1.00	0.05	1
Accuracy: 0.02 Macro Avg: 0.00 (Precision), 0.20 (Recall), 0.01 (F1-score) Weighted Avg: 0.00 (Precision), 0.02 (Recall), 0.00 (F1-score)				

the results are degraded. This can be fixed if we try to fine-tune the detection by trying out other percentage of best results rather than 2.5%.

5.2.2 Thresholding and Grouping Fine-Tuning

In continuation with the previous experiment, this section represents the results of the experiment when Thresholding and Grouping ranges are included. In the case of blocks, Both Threshold and Grouping are introduced simultaneously, as there is s higher chance of the same template to be detected twice. Thresholding parameter help in considering the top matches, and the grouping parameter take care if there are multiple detections for the same region. For block detection, Figure 5.1 – 5.3 represent the Thresholding and grouping matrix based on the accuracy scores. Then Table 5.19 – 5.21 represent the final classification report for block detection based on the fine-tuned threshold and grouping values for which the accuracy is the highest respectively for each algorithm. Similarly, for room detection, Figure 5.4 – 5.6 represents the Threshold graph based on the accuracy scores. Then Table 5.22 – 5.24 represents the final classification report for block detection based on the fine-tuned threshold and value for which the accuracy is the highest.

Block Detection

For Exp 1 we know that we have chosen the best 25% of the results for block detection. CCOEFF shows the best results at threshold of 0.70 and grouping range value of 1. CCORR shows best results at threshold of 0.86 and grouping range of 1. The threshold values are considered from 0.80 to 1.00 as a matter of fact that no actual accuracy was found near 0.75 mark for this algorithm as discussed in previous sections. Similarly, in SQDIFF, the best results are found at a threshold value of 0.3 and grouping range of 1. If all the three graphs are compared, then it is clearly visible that all of them attain a spike at a certain combination of threshold and grouping. Compared to grouping, threshold has a major impact on the algorithms as the transition between the accuracy values is gradual when considered horizontally

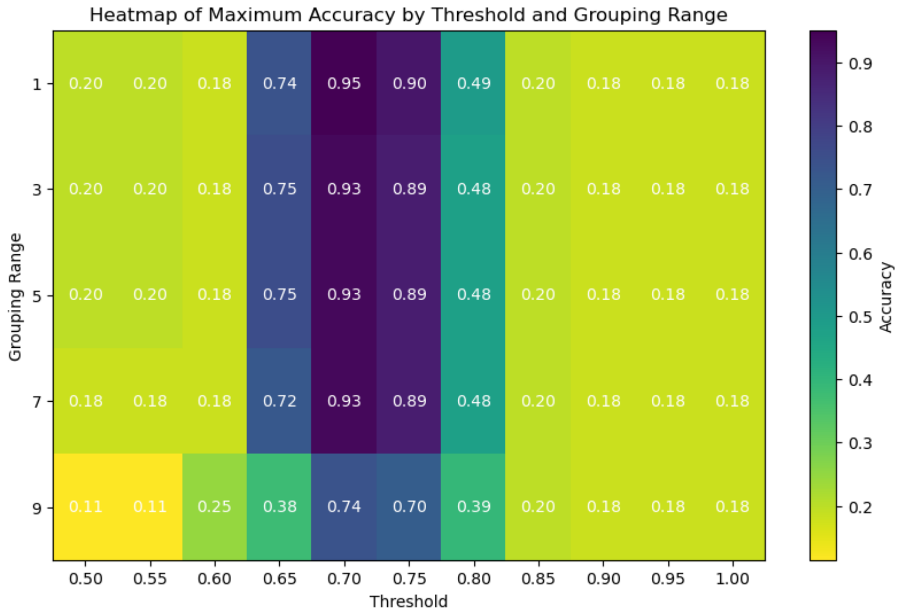

FIGURE 5.1: TM_CCOEFF_NORMED Heatmap : Threshold and Grouping

in the graph whereas if the values considered vertically, there is no much change observed.

TABLE 5.19: **TM_CCOEFF_NORMED — Block Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.92	1.00	0.96	11
Class 2	1.00	1.00	1.00	12
Class 3	0.92	1.00	0.96	12
Class 4	0.93	0.93	0.93	14
Class 5	1.00	0.83	0.91	12
Accuracy: 0.95 Macro Avg: 0.95 (Precision), 0.95 (Recall), 0.95 (F1-score) Weighted Avg: 0.95 (Precision), 0.95 (Recall), 0.95 (F1-score)				

The final metrics of each algorithm for block detection turns out to be good in every aspect. In comparison, CCOEFF is able to attain macro-precision of 0.95, macro-recall of 0.95, and macro-F1 of 0.95. Whereas, both CCORR and SQDIFF are able to achieve macro-precision of 0.97, macro-recall of 0.97, and macro-F1 of 0.97. On observation, it is visible that the threshold range considered for CCOEFF is 0.50 - 1.00 and for CCORR is 0.80 - 1.00, but for SQDIFF it is 0.05 - 0.50. The reason is, SQDIFF considered the square of the differences between the pixels values of the template and the region of the drawing under consideration. The lesser the difference, the better is the match. Hence, by specifying the threshold between 0.05 - 0.50, it is asked to take out results with the least difference. Unlike SQDIFF, for

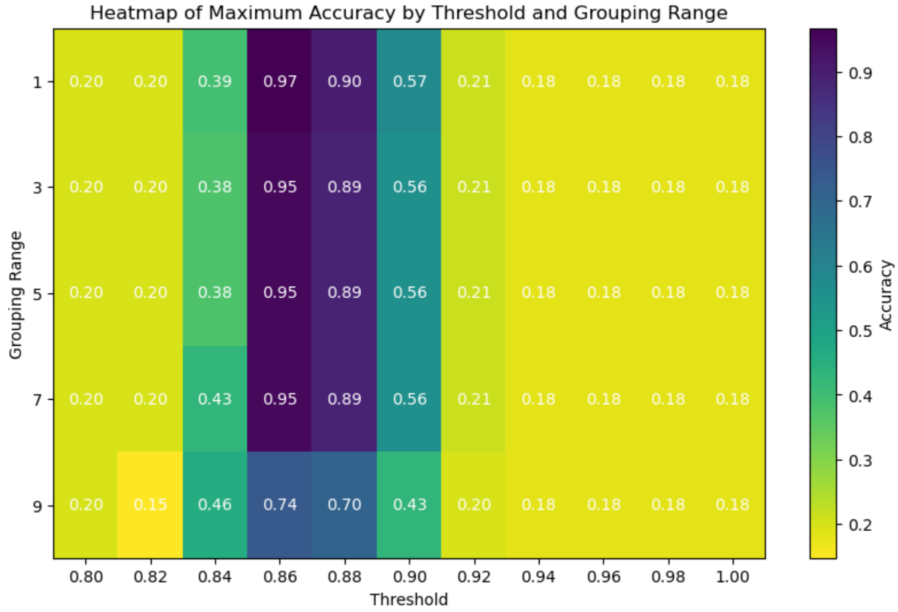

FIGURE 5.2: TM_CCORR_NORMED Heatmap : Threshold and Grouping

TABLE 5.20: **TM_CCORR_NORMED — Block Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.92	1.00	0.96	11
Class 2	1.00	1.00	1.00	12
Class 3	0.92	1.00	0.96	12
Class 4	1.00	0.93	0.96	14
Class 5	1.00	0.92	0.96	12
Accuracy: 0.97				
Macro Avg: 0.97 (Precision), 0.97 (Recall), 0.97 (F1-score)				
Weighted Avg: 0.97 (Precision), 0.97 (Recall), 0.97 (F1-score)				

CCOEFF and CCORR, higher values of threshold specifies better match.

Room Detection

For Exp 1, the best 25% of the results for block detection in case of CCOEFF have been taken and 1% & 2.5% for CCORR and SQDIFF respectively. In Exp 2, after considering rotation, flip and thresholding, CCOEFF shows the best results at threshold of 0.85 by attaining the accuracy of 70%, Means when the top 15% matches are considered. CCORR shows best results at threshold of 0.996 with an accuracy of 59%, means when top 0.4% results are considered. Similarly, in SQDIFF, the best results are found at a threshold value of 0.010 with an accuracy of 66%, means when top 1% results are considered. If all the three graphs are compared, then it is clearly visible that all of them attain a spike at a certain threshold value, just like the case on room detection. The reason for CCOEFF to show such great results

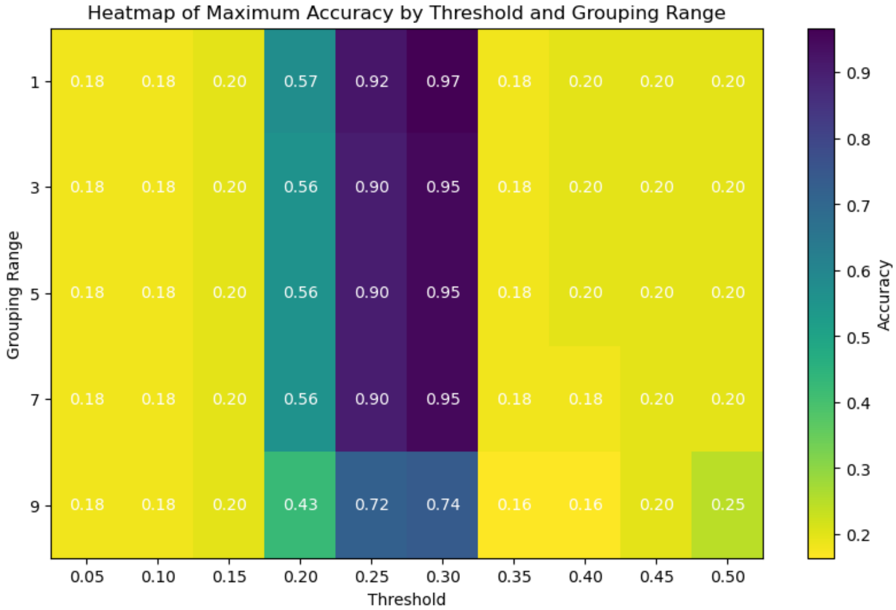

FIGURE 5.3: TM_SQDIFF_NORMED Heatmap : Threshold and Grouping

TABLE 5.21: **TM_SQDIFF_NORMED — Block Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.92	1.00	0.96	11
Class 2	1.00	1.00	1.00	12
Class 3	0.92	1.00	0.96	12
Class 4	1.00	0.93	0.96	14
Class 5	1.00	0.92	0.96	12
Accuracy: 0.97				
Macro Avg: 0.97 (Precision), 0.97 (Recall), 0.97 (F1-score)				
Weighted Avg: 0.97 (Precision), 0.97 (Recall), 0.97 (F1-score)				

even while considering the top 15% of the results is that it considers both strength and direction of similarity for every comparison. Where positive value of Correlation Coefficient means and highly similar match and a negative values concludes and dissimilar match. Unlike CCOEFF, CCORR, i.e. Cross Correlation, only computes the measure of similarity at each position which considers brightness and contrast of the pixels. Since both, the room templates and the drawings under consideration have a white base, the overall difference only arises by the black room edges, which makes the difference very close to each other and that's the reason the algorithm demands to consider the top 0.4% of the matches to do the distinguishing. Also for SQDIFF, as discussed formerly, it considers the square of the differences in the pixels values, the lower the difference, the better the result under consideration.

The final metrics clearly shows that CCOEFF has better results compared to CCORR and SQDIFF. In CCOEFF, 0.96 of recall for Class 1 states that out of

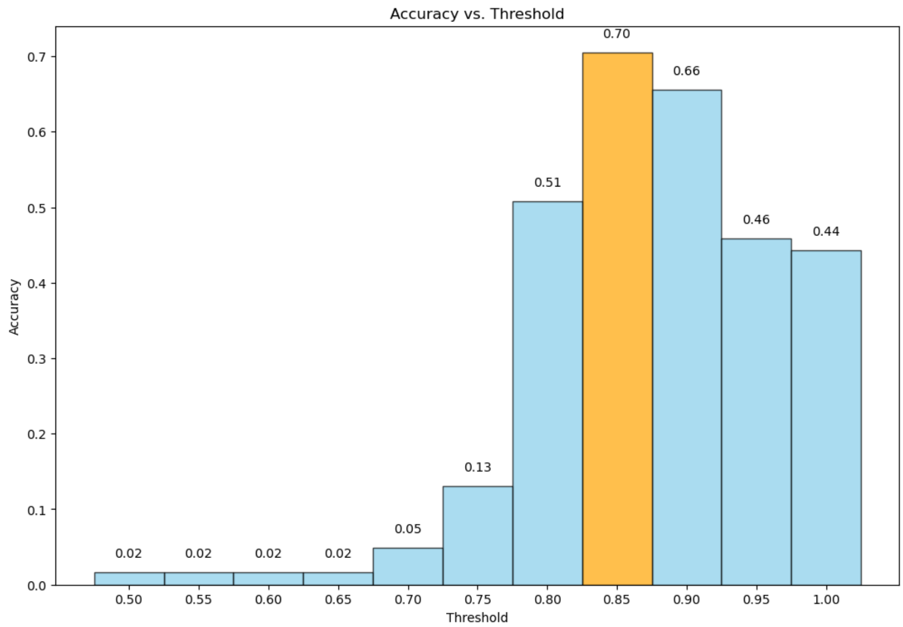

FIGURE 5.4: TM_CCOEFF_NORMED Threshold Graph

TABLE 5.22: **TM_CCOEFF_NORMED — Room Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.68	0.96	0.80	27
Class 2	0.81	0.65	0.72	20
Class 3	0.50	0.22	0.31	9
Class 4	0.67	0.50	0.57	4
Class 5	0.00	0.00	0.00	1
Accuracy: 0.70 Macro Avg: 0.53 (Precision), 0.47 (Recall), 0.48 (F1-score) Weighted Avg: 0.69 (Precision), 0.70 (Recall), 0.67 (F1-score)				

all the true one room drawings, the model is able to predict correctly for 96% of the times. Similarly, 65% accurately for 2 room blocks, 22% accurately for 3 room blocks, and 50% accurately for 4 room blocks. That means, the higher the room count, it is conventionally difficult to detect the accurate number of rooms. The reason behind this would be the two cases discussed in sec 5.1.2. Also, the proximity of the rooms is also a major point of discussion as sometimes when two rooms are close to each other there is a high chance that they are considered as one. This is the reason we haven't considered grouping parameter in the case of room detection, because it is good that it is avoided to combine the two different rooms together. Also in terms of Precision, as shown in fig 5.22, Out of all the predictions made by the CCOEFF algorithm. 68% times it is said correctly as 1 room block, 81% times for the two room block, 50% and 67% times correctly for the 3 and 4 rooms block respectively. Since there is only one instance of having a 5 room block, it is understandable currently to not consider that part. According to

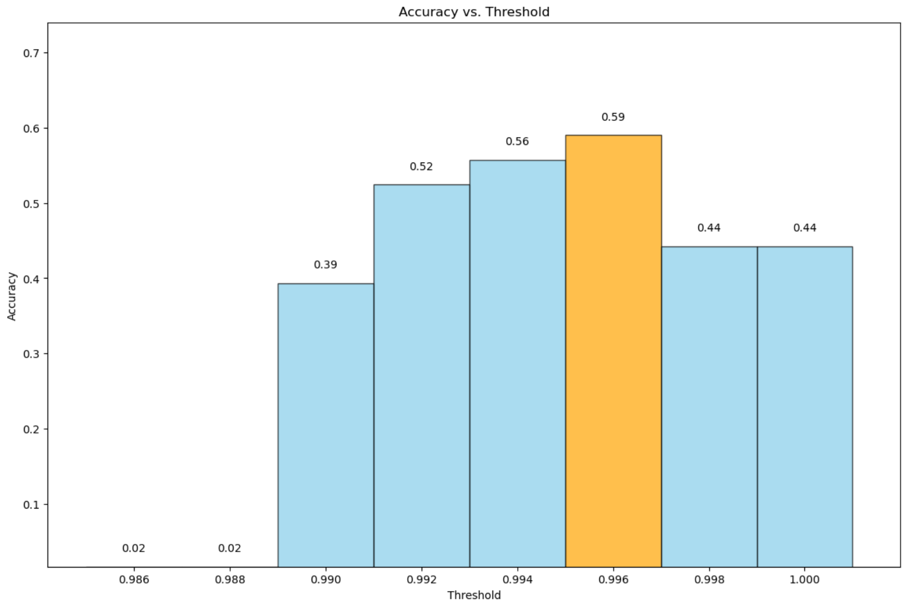

FIGURE 5.5: TM_CCORR_NORMED Threshold Graph

TABLE 5.23: **TM_CCORR_NORMED — Room Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.52	1.00	0.68	27
Class 2	1.00	0.45	0.62	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.00	0.00	0.00	1
Accuracy: 0.59				
Macro Avg: 0.30 (Precision), 0.29 (Recall), 0.26 (F1-score)				
Weighted Avg: 0.56 (Precision), 0.59 (Recall), 0.51 (F1-score)				

figure 5.23, CCORR does not show comparatively good results as CCOEFF. Based on individual precision values, 52% of the times model is correct when predicted drawing to have 1 room and 100% of the times in case of 2 rooms. The Recall of Class 2 is 0.45 says that out of actual 2 room drawings, 45% of the times model is able to detect them accurately. There is no considerable detection for the rest of the classes. In case of SQDIFF, the results are almost close to the performance of CCOEFF for Classes 1 and 2, but there is no considerable results for other classes, the reason remains the same discussed in section 5.1.2 s CASE I and II. Yet, this model is considerable if we talk about the drawings having 1 and 2 room blocks. On comparison, the macro scores of precision, recall, and F1 scores for CCOEFF are found to be beyond 47% mark. followed by SQDIFF with beyond 29% and then CCORR with beyond 26%. Since macro-F1-scores are the harmonic mean of the macro-precision and macro-recall, and it will only increase when both the values increase simultaneously. Also based on our proposed metric DEI, it tries to find the overall better algorithm in the detection of both number of block and

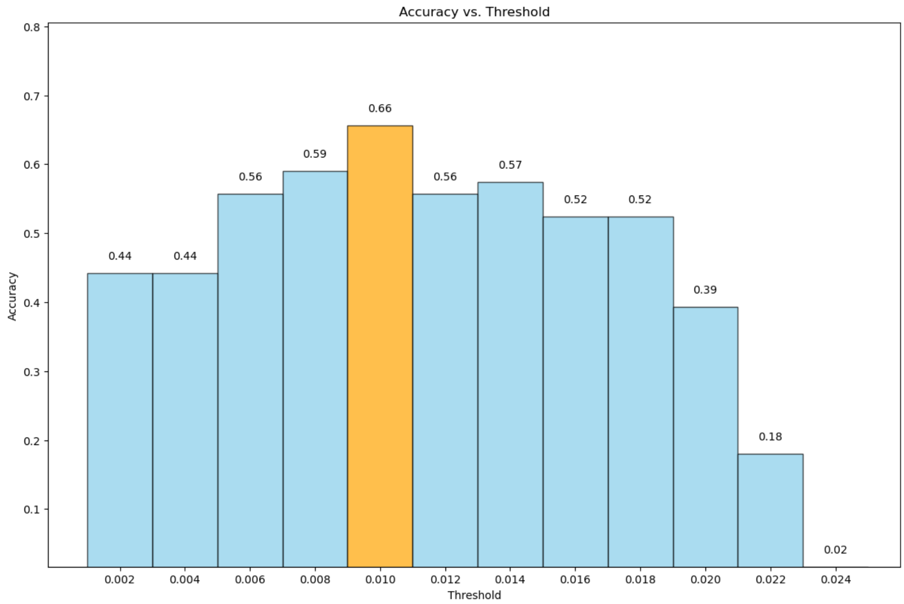

FIGURE 5.6: TM_SQDIFF_NORMED Threshold Graph

TABLE 5.24: **TM_SQDIFF_NORMED — Room Detection Final Classification Report**

Class	Metrics			Support
	Precision	Recall	F1-score	
Class 1	0.63	1.00	0.77	27
Class 2	0.81	0.65	0.72	20
Class 3	0.00	0.00	0.00	9
Class 4	0.00	0.00	0.00	4
Class 5	0.00	0.00	0.00	1
Accuracy: 0.66				
Macro Avg: 0.29 (Precision), 0.33 (Recall), 0.30 (F1-score)				
Weighted Avg: 0.54 (Precision), 0.66 (Recall), 0.58 (F1-score)				

number of rooms. The overall score of the DEI metric rises only when the individual scores of the macro-F1 scores of block and room detection rises. And the individual macro-F1 will attain higher values when there are accurate detections of both the features resulting in higher values of individual macro-precision and macro-recall The sequence of algorithm consideration can be seen as shown in table 5.25. Hence, based on that criteria, CCOEFF(0.63) ace the list followed by SQDIFF(0.45) and CCORR(0.41). The higher value of CCOEFF compared to other two algorithms claims that the algorithm has more accurate detections when both the parameters are considered simultaneously. Had it been the case when only block detection is considered, based on table 5.25 COEFF is behind by 0.02% which is a negligible difference when the test set size is considered.

TABLE 5.25: Detection Effectiveness Index

	Macro-F1(Block)	**Macro-F1(Room)**	**DEI**
TM_CCOEFF_NORMED	0.95	0.48	0.6377
TM_CCORR_NORMED	0.97	0.26	0.4100
TM_SQDIFF_NORMED	0.97	0.30	0.4582

5.2.3 Validation

TABLE 5.26: Block Detection (Validation)

Algorithm	Macro-Precision	Macro-Recall	Macro-F1	Accuracy
TM_CCOEFF_NORMED	0.84	0.82	0.79	0.80
TM_CCORR_NORMED	0.90	0.86	0.84	0.85
TM_SQDIFF_NORMED	0.73	0.73	0.73	0.75

TABLE 5.27: Room Detection (Validation)

Algorithm	Macro-Precision	Macro-Recall	Macro-F1	Accuracy
TM_CCOEFF_NORMED	0.85	0.81	0.82	0.85
TM_CCORR_NORMED	0.60	0.58	0.55	0.65
TM_SQDIFF_NORMED	0.58	0.58	0.54	0.65

TABLE 5.28: Detection Effectiveness Index(Validation)

Algorithm	DEI
TM_CCOEFF_NORMED	**0.85**
TM_CCORR_NORMED	0.60
TM_SQDIFF_NORMED	0.58

Table 5.26 and 5.27 represents the results obtained by the algorithms with the parametric values they have performed best. In case of blocks CCORR has the highest macro-precison of 0.90 which means majority of the times the predicted class by the algorithm is correct. Similarly highest macro-recall of 0.86 means that out of all the algorithms, CCORR predicts the correct class better when compared. Eventually, gets the highest macro-F1 scores and the accuracy scores of 0.84 & 0.85 respectively. Since, there are 20 instances for validation, every drawing constitues for 5% when classified correctly. Accordingly, in case of small icon detections, when the algorithms are executed on their personal best parameter values, CCORR attains the maximum accuracy of 85% followed by CCOEFF and SQDIFF with 80% and 75% respectively. In case of rooms, CCOEFF is the clear winner in terms of macro-precision with 0.85, macro-recall with 0.81, evaluating to macro-F1 of 0.82 and the accuracy of 85%. The reason why CCOEFF performs better is because it computes the correlation coefficient which takes into account both the strength and direction of similarity(+ve value for positive similarity and -ve for negative similarity). Since, the room templates are atleast 500x bigger than the block icon, computing the correlation coefficient makes it more accurate to detect the similarity in template. Hence, CCOEFF performs better than CCORR and SQDIFF in case of rooms. When the DEI score is calculated, CCOEFF has the score of 0.85 which is atleast

33% more than the DEI score of CCORR and SQDIFF. Hence, giving the clear winner.

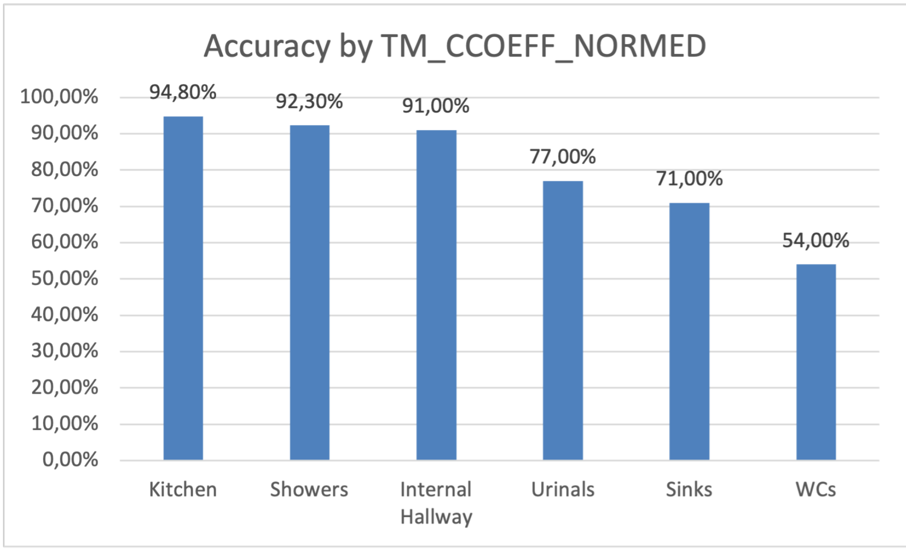

FIGURE 5.7: TM_CCOEFF_NORMED Accuracy Graph

To support the correctness of the finding, the CCOEFF is used to detect few other icons and patterns from the drawings. To execute this validation experiment, all the drawings including the design-test and the validation constituted around 116 drawings. As shown in figure 5.7, CCOEFF is able to kitchen with 94.80%, showers with 92.30%, internal hallway with 91.00%, urinals with 77.00%, sinks with 71% and WCs with 54% accuracy. The accuracy for various icons suggest that, out of all the drawings, the model is able to extract the exact number of the quantity in the drawings correctly. Hence, all the false negatives and false positives also accounts for the cases where the icons are detected in the drawing but the exact number of matches are not found. Internal hallway is an exception as the outcome is boolean in nature, i.e. internal hallway present or not.

Based on the results, discussions, and associated validation steps, the findings provide a strong foundation to use this research for practical purposes. It can be used to create a proper metadata model for the drawings. The metadata can then be used to retrieve the drawings based on certain properties. In scenarios, whenever a stakeholder is interested in finding a drawing with some unique requirements, a mapping algorithm can be implemented to map those customer requirements with the metadata. The best-matched results can then be displayed without any hassle. Architectural industries can use this to gather ideas from past projects, which would help to expedite their designing process. Additionally, the findings can help in estimating the cost of the project based on the various facilities available in a drawing. This information could be utilized by project managers and financial teams to more accurately estimate project costs upfront and allocate budgets accordingly. This could lead to more efficient resource allocation, better financial planning, and reduced risks of cost overruns during the execution phase of architectural projects.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

This section answers the research questions and their sub parts that were designed in the Chapter 1. Based on the Experiments and their results, the answers to those research questions are as follows:

- **RQ 1. What are the optimal detection techniques for extracting details from construction drawings?**

  From Section 5.1, it can be concluded that out of 5 detection algorithms under consideration, CCOEFF, CCORR, SQDIFF are the preferred algorithms when the use case is similar to what is presented, i.e. when the agenda is to find a smaller piece of visual information on a large image and also want to count the number of occurrences. In case of blocks, at a certain threshold value where only best 25% of the matching results were considered, CCOEFF and SQDIFF show the macro-F1 scores of 0.83 and 0.84 respectively. CCORR couldn't beat them, but later after fine-tuning, it gave a tough competition. In case of rooms, CCOEFF, CCORR and SQDIFF again were the best options to consider from the considered 5 algorithms. There is not a clear winner for the room detection task until later when the fine-tuning is performed to get better results.

- **RQ 2. What are the approaches to fine-tune the algorithm for efficient results?**

  Out of various fine-tuning methods discussed in section 2.2.2. Considering Rotation and Flip, followed by introducing Thresholding and Grouping as tested in the section 5.2 did turn out in favour of improving the performance of the algorithms under consideration. In case of blocks, when only Rotation and Flip is introduced, CCOEFF and SQDIFF showed improvement as they attained the macro-F1 scores of 0.90 and 0.92 respectively. Later, when thresholding and grouping are also introduced, there is a shift noticed in the rankings. As both CCORR and SQDIFF showed, the macro-F1 scores of 0.97 followed by CCOEFF with the score of 0.95. In case of blocks, after introducing only rotation and flip, no significant improvement is noticed in any algorithm whereas when thresholding is introduced along with it, a positive shift is noticed. CCOEFF is able to achieve the macro-F1 scores of 0.48(previously 0.14). Followed by SQDIFF and CCORR with the macro-F1 scores of 0.30(previously 0.08) and 0.26(previously 0.20) respectively.

- **RQ 3. What are the evaluation parameters to measure the success of the extraction of metadata from construction drawings?**

  From section 4.3, it can be concluded that the considered metrics would be the good option in the current context to analyse the results. With the individual scores of Precision, Recall, and F1 for every class, it gives us the performance of how the algorithm performs at the individual level. Micro scores are considered in order to give equal weightage to each class, even there is an unequal distribution of classes in the dataset. The proposed metric termed as DEI gives us the overall score to evaluate the subtasks to answer the best preferred algorithm.

- **RQ. Is the extraction of metadata from construction drawings feasible?**

  From Section 2.1.1 and 2.1.2 discusses the feasibility of extracting metadata from the construction drawings. Overall, based on our experiments stated in section 5.1 and 5.2 it is validated that, given the construction drawing, it is possible to extract the metadata from the construction drawing with the help of detection algorithms which are not specifically trained over large amount of labelled drawings. Hence, providing a constructive validation of the possibility of good performance without any overhead involvement of time and computational resources.

6.2 Scientific Contribution

This research represents a significant contribution to the field by focusing on the development and evaluation of algorithms that prioritize cost-effectiveness while delivering robust results. By carefully reviewing and assessing algorithms such as CCORR, CCOEFF, and SQDIFF, the study sheds light on their viability in scenarios where resource constraints are a concern.

Moreover, the research addresses a common challenge faced by mid-sized companies, namely the lack of infrastructure for real-time data capture and storage suitable for model execution and training. Unlike traditional machine learning algorithms, which often require thousands of instances for meaningful results[65], this study demonstrates the feasibility of achieving significant outcomes even with limited data availability.

Furthermore, the findings of this research underscore the potential of image processing algorithms, particularly CCORR, CCOEFF, and SQDIFF, in detecting and extracting metadata from construction drawings. Through optimization techniques such as rotation, flip, thresholding, and grouping, these algorithms exhibited enhanced performance, particularly in scenarios where drawings follow generalized patterns or templates. The optimization efforts yielded notably positive results, indicating the effectiveness of these methods in improving the accuracy and efficiency of metadata extraction processes. Additionally, the CRISP-DM methodology facilitated the iterative nature of the project, allowing for continuous refinement and optimization based on feedback and evaluation results.

In essence, this research contributes to advancing the understanding and application of image processing algorithms in real-world contexts characterized by resource constraints. By demonstrating the efficacy of cost-effective algorithms and optimization strategies, the study offers valuable insights for practitioners and researchers alike, paving the way for enhanced efficiency and innovation in industries reliant on image data analysis.

6.3 Future Works

The study highlights several limitations and proposes avenues for future research to enhance its findings. Foremost among these limitations is the restricted size of the dataset. The experiments are strategically conducted by making sure that the design set and the test set have similar distribution of classes. To improve the study's comprehensiveness and robustness, increasing the dataset's size is paramount. This might result in improving the results in room detections multifold. The second limitation is the computational resources. Due to privacy reasons, it was not allowed to use any third party applications and cloud computing facility, which could have increased the throughput time. Unavailability of large set of labelled data was also something needs to be focussed on and that's what the challenge was all about its feasibility. To further take this research based on its results, there is a scope of proposing a semantic model for the construction and architectural drawings which can also help in mapping the metadata acquired by the drawings to map it to the customer queries based on their requirement.

Bibliography

[1] Amal Allagui, Imen Belhadj, Nizar Aifaoui, Moncef Hammadi, and Jean-Yves Choley. A computer aided design - system engineering interoperability by enriching computer aided design database with functional information. In *2021 18th International Multi-Conference on Systems, Signals Devices (SSD)*, pages 1452–1458, 2021. doi:10.1109/SSD52085.2021.9429313.

[2] Mehrdad Arashpour, Tuan Ngo, and Heng Li. Scene understanding in construction and buildings using image processing methods: A comprehensive review and a case study. *Journal of Building Engineering*, 33:101672, 2021. URL: https://www.sciencedirect.com/science/article/pii/S235271022031915X, doi:10.1016/j.jobe.2020.101672.

[3] Subhadeep Bagui and Kuan-Ching Li. Resampling imbalanced data for network intrusion detection datasets. *Journal of Big Data*, 8(1):6, 2021. doi:10.1186/s40537-020-00390-x.

[4] Manju Bansal, Mukesh Kumar, and Munish Kumar. 2d object recognition: A comparative analysis of sift, surf, and orb feature descriptors. *Multimedia Tools and Applications*, 80:18839–18857, 2021. doi:10.1007/s11042-021-10646-0.

[5] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3951 LNCS:404 – 417. Cited by: 8688; All Open Access, Bronze Open Access, Green Open Access. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-33745827782&doi=10.1007%2f11744023_32&partnerID=40&md5=625abb60e9dbc6e0a2cf64adee7d3fd0, doi:10.1007/11744023_32.

[6] Thomas Blaschke, Charles Burnett, and Anssi Pekkarinen. Image segmentation methods for object-based analysis and classification. In *Remote sensing image analysis: Including the spatial domain*, pages 211–236. Springer, 2004.

[7] Paula Branco, Luís Torgo, and Rita P Ribeiro. A survey of predictive modeling on imbalanced domains. *ACM computing surveys (CSUR)*, 49(2):1–50, 2016.

[8] James J. Burke and John G. Thunen. Progress in analog image processing. Technical report, 1968. URL: http://hdl.handle.net/10150/621628.

[9] Andrew David Calway. *The multiresolution Fourier transform: a general purpose tool for image analysis*. PhD thesis, University of Warwick, 1989.

[10] Gabriella Castellano and Gennaro Vessio. Deep learning approaches to pattern extraction and recognition in paintings and drawings: an overview. *Neural Computing and Applications*, 33:12263–12282, 2021. doi:10.1007/s00521-021-05893-z.

[11] Yunlei Chen. Perspective data analysis and mining algorithm for interior art design from the perspective of virtual metadata-assisted 3d modeling.

[12] Guoming Chu, Yan Peng, and Xuhong Luo. Algd-orb: An improved image feature extraction algorithm with adaptive threshold and local gray difference. *Plos one*, 18(10):e0293111, 2023.

[13] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 113–123, 2019. URL: https://openaccess.thecvf.com/content_CVPR_2019/papers/Cubuk_AutoAugment_Learning_Augmentation_Strategies_From_Data_CVPR_2019_paper.pdf, doi:10.1109/CVPR.2019.00020.

[14] Oladipo Dare-Abel, Obioha Uwakonye, and Akunnaya Opoko. The effect of computer aided design on architecture students' creativity and enthusiasm. 03 2016. doi:10.21125/inted.2016.0984.

[15] Sanjib Das, Jonti Saikia, Soumita Das, and NA Goni. Comparative study of different noise filtering techniques in digital images. *International Journal of Engineering Research and General Science*, 3(5):180–191, 2015.

[16] Terrance Devries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. *ArXiv*, abs/1708.04552, 2017. URL: https://api.semanticscholar.org/CorpusID:23714201.

[17] Zhiwen Fan, Lingjie Zhu, Honghua Li, Xiaohao Chen, Siyu Zhu, and Ping Tan. Floorplancad: A large-scale computer aided drawing dataset for panoptic symbol spotting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10128–10137, October 2021.

[18] Lin Fu, Jun Zhu, Wei-lian Li, Ji-gang You, and Zhang-yu Hua. Fast estimation method of volumes of landslide deposit by the 3d reconstruction of smartphone images. *Landslides*, 18(9):3269 – 3278, 2021. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85106435139&doi=10.1007%2fs10346-021-01702-9&partnerID=40&md5=304817e2fda05a2fc2255ca37b2673a0, doi:10.1007/s10346-021-01702-9.

[19] Joseph Yossi Gil and Ron Kimmel. Efficient dilation, erosion, opening, and closing algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12):1606–1617, 2002.

[20] Dorothy Gors, Jeroen Put, Bram Vanherle, Maarten Witters, and Kris Luyten. Semi-automatic extraction of digital work instructions from computer aided design models. *Procedia CIRP*, 97:39–44, 2021. 8th CIRP Conference of Assembly

Technology and Systems. URL: https://www.sciencedirect.com/science/article/pii/S2212827120314190, doi:10.1016/j.procir.2020.05.202.

[21] Megha Goyal. Morphological image processing. *IJCST*, 2(4):59, 2011.

[22] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3012–3021, 2020.

[23] Shagun Gupta, Shubham Mahajan, and Amit Kant Pandit. A review on image processing techniques. In *2020 12th International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 20–24, Sep. 2020. doi:10.1109/CICN49253.2020.9242606.

[24] S. Indhumathi and J.Christopher Clement. Comparison of feature detection using sift and orb detector. In *TENCON 2022 - 2022 IEEE Region 10 Conference (TENCON)*, pages 1–3, 2022. doi:10.1109/TENCON55691.2022.9977522.

[25] Tao Jiang and Yuchen Xue. An improved orb feature point extraction algorithm based on image enhancement and adaptive threshold. In *2023 IEEE 7th Information Technology and Mechatronics Engineering Conference (ITOEC)*, volume 7, pages 1233–1238, 2023. doi:10.1109/ITOEC57671.2023.10291977.

[26] Manjunath Jogin, MS Madhulika, GD Divya, RK Meghana, S Apoorva, et al. Feature extraction using convolution neural networks (cnn) and deep learning. In *2018 3rd IEEE international conference on recent trends in electronics, information & communication technology (RTEICT)*, pages 2319–2323. IEEE, 2018.

[27] Alexey Kashevnik, Nikolay Shilov, Nikolay Teslya, Fudail Hasan, Andrey Kitenko, Veronika Dukareva, Marat Abdurakhimov, Alexander Zingarevich, and Dmitry Blokhin. An approach to engineering drawing organization: Title block detection and processing. *IEEE Access*, pages 1–1, 2023. doi:10.1109/ACCESS.2023.3244603.

[28] Jung Rok Kim and Jae Wook Jeon. Real-time implement orb algorithm in fpga : ofast with harris corner and rbrief algorithm. In *2021 IEEE Region 10 Symposium (TENSYMP)*, pages 1–4, 2021. doi:10.1109/TENSYMP52854.2021.9551004.

[29] Sung Kim and Riley Casper. Applications of convolution in image processing with matlab. *University of Washington*, pages 1–20, 2013.

[30] Eivind Kristoffersen, Oluseun Omotola Aremu, Fenna Blomsma, Patrick Mikalef, and Jingyue Li. Exploring the relationship between data science and circular economy: An enhanced crisp-dm process model. In Ilias O. Pappas, Patrick Mikalef, Yogesh K. Dwivedi, Letizia Jaccheri, John Krogstie, and Matti Mäntymäki, editors, *Digital Transformation for a Sustainable Society in the 21st Century*, pages 177–189, Cham, 2019. Springer International Publishing.

[31] Raghu Raj Prasanna Kumar, Suresh Muknahallipatna, and John McInroy. An approach to parallelization of sift algorithm on gpus for real-time applications. *Journal of Computer and Communications*, 4(17):18–50, 2016.

[32] Jong-Sen Lee. Digital image enhancement and noise filtering by use of local statistics. *IEEE transactions on pattern analysis and machine intelligence*, (2):165–168, 1980.

[33] Jong-Sen Lee. Digital image smoothing and the sigma filter. *Computer vision, graphics, and image processing*, 24(2):255–269, 1983.

[34] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017. URL: https://api.semanticscholar.org/CorpusID:47252984.

[35] Jianqing Liu and Yee-Hong Yang. Multiresolution color image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(7):689–700, 1994.

[36] Jiazhen LU, Maoqing HU, Jing DONG, Songlai HAN, and Ang SU. A novel dense descriptor based on structure tensor voting for multi-modal image matching. *Chinese Journal of Aeronautics*, 33(9):2408–2419, 2020. URL: https://www.sciencedirect.com/science/article/pii/S1000936120300534, doi:10.1016/j.cja.2020.02.002.

[37] Lichun Mei, Caiyun Wang, Huaiye Wang, Yuanfu Zhao, Jun Zhang, and Xiaoxia Zhao. Fast template matching in multi-modal image under pixel distribution mapping. *Infrared Physics Technology*, 127:104454, 2022. URL: https://www.sciencedirect.com/science/article/pii/S1350449522004352, doi:10.1016/j.infrared.2022.104454.

[38] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7):3523–3542, 2021.

[39] Zhao Mm, Sun Jian, Sun Dihua, and Tang Yi. Fast tunnel monitoring video stabilization method based on improved orb feature. In *2018 Chinese Control And Decision Conference (CCDC)*, pages 5856–5861, 2018. doi:10.1109/CCDC.2018.8408155.

[40] Alhassan Mumuni and Fuseini Mumuni. Data augmentation: A comprehensive survey of modern approaches. *Array*, 16:100258, 2022. URL: https://www.sciencedirect.com/science/article/pii/S2590005622000911, doi:10.1016/j.array.2022.100258.

[41] Federica Pascucci. How digital technologies reshape marketing: evidence from a qualitative investigation - Italian Journal of Marketing — link.springer.com. https://link.springer.com/article/10.1007/s43039-023-00063-6#citeas, 2023. [Accessed 23-11-2023].

[42] Richard Alan Peters. A new algorithm for image noise reduction using mathematical morphology. *IEEE transactions on Image Processing*, 4(5):554–568, 1995.

[43] Najeeb Ur Rehman Malik, Awais Gul Airij, Sheeraz Ahmed Memon, Yasmeen Naz Panhwar, Syed A.R. Abu-Bakar, and Mohamed A. El-Khoreby. Performance comparison between surf and sift for content-based image retrieval. In *2019 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 214–218, 2019. doi:10.1109/ICSIPA45851.2019.8977732.

[44] Jadwiga Rogowska. Overview and fundamentals of medical image segmentation. *Handbook of medical imaging, processing and analysis*, pages 69–85, 2000.

[45] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International Conference on Computer Vision*, pages 2564–2571, 2011. doi:10.1109/ICCV.2011.6126544.

[46] Rafael Sacks, Charles Eastman, Ghang Lee, and Paul Teicholz. *BIM Handbook: A Guide to Building Information Modeling for Owners, Designers, Engineers, Contractors, and Facility Managers*. 08 2018. doi:10.1002/9781119287568.

[47] Prasanna K Sahoo, SAKC Soltani, and Andrew KC Wong. A survey of thresholding techniques. *Computer vision, graphics, and image processing*, 41(2):233–260, 1988.

[48] Jignesh Sarvaiya, Suprava Patnaik, and Salman Bombaywala. Image registration by template matching using normalized cross-correlation. *Advances in Computing, Control, and Telecommunication Technologies, International Conference on*, 0:819–822, 12. doi:10.1109/ACT.2009.207.

[49] Beate Scheibel, Juergen Mangler, and Stefanie Rinderle-Ma. Extraction of dimension requirements from engineering drawings for supporting quality control in production processes. *Computers in Industry*, 129:103442, 2021. URL: https://www.sciencedirect.com/science/article/pii/S016636152100049X, doi:10.1016/j.compind.2021.103442.

[50] Shammi Shawal, Muhammad Shoyab, Suraiya Begum, et al. Fundamentals of digital image processing and basic concept of classification. *International Journal of Chemical and Process Engineering Research*, 1(6):98–108, 2014.

[51] Frank Y Shih. *Image processing and pattern recognition: fundamentals and techniques*. John Wiley & Sons, 2010.

[52] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(60), 2019. doi:10.1186/s40537-019-0197-0.

[53] Li Shuhua and Guo Gaizhi. The application of improved hsv color space model in image processing. In *2010 2nd International Conference on Future Computer and Communication*, volume 2, pages V2–10. IEEE, 2010.

[54] Montree Srikham, Charnchai Pluempitiwiriyawej, and Tawee Chanwimaluang. Comparison of dense matching algorithms in noisy image. In *Second International Conference on Digital Image Processing*, volume 7546, pages 434–440. SPIE, February 2010.

[55] J.S. Sujin and S. Sophia. High-performance image forgery detection via adaptive sift feature extraction for low-contrast or small or smooth copy–move region images. *Soft Computing*, 28(1):437 – 445, 2024. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85152554281&doi=10.1007%2fs00500-023-08209-6&partnerID=40&md5=34eb0062e0a75e59d044fe430b5ec0ff, doi:10.1007/s00500-023-08209-6.

[56] Priya Vasanth Sundara Rajan and A Lenin Fred. An efficient compound image compression using optimal discrete wavelet transform and run length encoding techniques. *Journal of Intelligent Systems*, 28(1):87–101, 2019.

[57] Sabine Süsstrunk, Robert Buckley, and Steve Swen. Standard rgb color spaces. In *Proc. IS&T;/SID 7th Color Imaging Conference*, volume 7, pages 127–134, 1999.

[58] Richard Swedberg. Exploratory research. *The production of knowledge: Enhancing progress in social science*, pages 17–41, 2020.

[59] Paul Teicholz. Bim for facility managers. 12 2018. doi:10.1002/9781119572633.

[60] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Herv'e J'egou. Fixing the train-test resolution discrepancy: Fixefficientnet. *ArXiv*, abs/2003.08237, 2020. URL: https://api.semanticscholar.org/CorpusID:212747706.

[61] P Umesh. Image processing in python. *CSI Communications*, 23(2), 2012.

[62] Ngo Van Quan, Duong Dinh Luyen, Phan Huy Anh, Pham Thi Hoai Thu, Nguyen Chi Thanh, and Vu Duc Thai. Speeding up sift for faster feature-based uav visual localization. *Lecture Notes in Networks and Systems*, 848:83 – 90, 2024. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85181984552&doi=10.1007%2f978-3-031-50818-9_11&partnerID=40&md5=fc54a90ef9f3ce97e23aba22f123f158, doi:10.1007/978-3-031-50818-9_11.

[63] Peter C. Verhoef, Thijs Broekhuizen, Yakov Bart, Abhi Bhattacharya, John Qi Dong, Nicolai Fabian, and Michael Haenlein. Digital transformation: A multidisciplinary reflection and research agenda. *Journal of Business Research*, 122:889–901, 2021. URL: https://www.sciencedirect.com/science/article/pii/S0148296319305478, doi:10.1016/j.jbusres.2019.09.022.

[64] Peter Vogt, Kurt H Riitters, Christine Estreguil, Jacek Kozak, Timothy G Wade, and James D Wickham. Mapping spatial patterns with morphological image processing. *Landscape ecology*, 22:171–177, 2007.

[65] Lidong Wang. Machine learning in big data. *International Journal of Advances in Applied Sciences*, 4:117, 12 2015. doi:10.11591/ijaas.v4.i4.pp117-123.

[66] Peipei Wang. Impact of brand marketing strategies based on consumer purchase intention mining. *Computer-Aided Design and Applications*, 21(S12):205 – 219, 2024. Cited by: 0; All Open Access, Bronze Open Access. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85176808501&doi=10.14733%2fcadaps.2024.S12.205-219&partnerID=40&md5=335f20026d91e45238b62e978dd2f20e, doi:10.14733/cadaps.2024.S12.205-219.

[67] David Waterworth, Subbu Sethuvenkatraman, and Quan Z. Sheng. Advancing smart building readiness: Automated metadata extraction using neural language processing methods. *Advances in Applied Energy*, 3:100041, 2021. URL: https://www.sciencedirect.com/science/article/pii/S2666792421000330, doi:10.1016/j.adapen.2021.100041.

[68] Wei-Yi Wei. An introduction to image compression. *National Taiwan University, Taipei, Taiwan, ROC*, 2008.

[69] Yair Wiseman. The still image lossy compression standard-jpeg. In *Encyclopedia of Information Science and Technology, Third Edition*, pages 295–305. IGI global, 2015.

[70] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Zhou Ren, and Alan Loddon Yuille. Mitigating adversarial effects through randomization. *ArXiv*, abs/1711.01991, 2017. URL: https://api.semanticscholar.org/CorpusID:3526769.

[71] Yunyun Yang, Wenjing Jia, and Boying Wu. Simultaneous segmentation and correction model for color medical and natural images with intensity inhomogeneity. *The Visual Computer*, 36:717–731, 2020.

[72] Lifan Yao, Hao Feng, Yiqun Zhu, Zhiguo Jiang, Danpei Zhao, and Wenquan Feng. An architecture of optimised sift feature detection for an fpga implementation of an image matcher. In *2009 International Conference on Field-Programmable Technology*, pages 30–37, 2009. doi:10.1109/FPT.2009.5377651.

[73] Shiping Zhu, Xi Xia, Qingrong Zhang, and Kamel Belloulata. An image segmentation algorithm in image processing based on threshold segmentation. In *2007 third international IEEE conference on signal-image technologies and internet-based system*, pages 673–678. IEEE, 2007.

[74] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning transferable architectures for scalable image recognition. 2018. URL: https://openaccess.thecvf.com/content_cvpr_2018/papers/Zoph_Learning_Transferable_Architectures_CVPR_2018_paper.pdf.

[75] Yaobin Zou, Fangmin Dong, Bangjun Lei, Lulu Fang, and Shuifa Sun. Image thresholding based on template matching with arctangent hausdorff distance

measure. *Optics and Lasers in Engineering*, 51:600–609, 05 2013. doi:10.1016/j.optlaseng.2012.12.016.