Using artificial intelligence to find design errors in the engineering drawings

Citation for the original published paper (version of record):

Dzhusupova, R., Banotra, R., Bosch, J. et al (2023). Using artificial intelligence to find design errors in the engineering drawings. Journal of Software: Evolution and Process, 35(12).
http://dx.doi.org/10.1002/smr.2543

Received: 23 September 2022 | Revised: 22 December 2022 | Accepted: 21 January 2023

DOI: 10.1002/smr.2543

RESEARCH ARTICLE - TECHNOLOGY

Journal of Software: Evolution and Process WILEY

Using artificial intelligence to find design errors in the engineering drawings

Rimma Dzhusupova[1] | Richa Banotra[2] | Jan Bosch[3] | Helena Holmström Olsson[4]

[1]Electrical, Instrumentation, Control & Safety Systems, McDermott, The Hague, The Netherlands

[2]Instrumentation, Control & Safety Systems, McDermott, The Hague, The Netherlands

[3]Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden

[4]Computer Science and Media Technology, Malmö University Malmö, Malmö, Sweden

Correspondence
Rimma Dzhusupova, Electrical, Instrumentation, Control & Safety Systems, McDermott, The Hague, The Netherlands.
Email: rdzhusupova@mcdermott.com

Funding information
This research was supported by McDermott.

Abstract

Artificial intelligence is increasingly becoming important to businesses because many companies have realized the benefits of applying machine learning (ML) and deep learning (DL) in their operations. ML and DL have become attractive technologies for organizations looking to automate repetitive tasks to reduce manual work and free up resources for innovation. Unlike rule-based automation, typically used for standardized and predictable processes, machine learning, especially deep learning, can handle more complex tasks and learn over time, leading to greater accuracy and efficiency improvements. One of such promising applications is to use AI to reduce manual engineering work. This paper discusses a particular case within McDermott where the research team developed a DL model to do a quality check of complex blueprints. We describe the development and the final product of this case—AI-based software for the engineering, procurement, and construction (EPC) industry that helps to find the design mistakes buried inside very complex engineering drawings called piping and instrumentation diagrams (P&IDs). We also present a cost-benefit analysis and potential scale-up of the developed software. Our goal is to share the successful experience of AI-based product development that can substantially reduce the engineering hours and, therefore, reduce the project's overall costs. The developed solution can also be potentially applied to other EPC companies doing a similar design for complex installations with high safety standards like oil and gas or petrochemical plants because the design errors it captures are common within this industry. It also could motivate practitioners and researchers to create similar products for the various fields within engineering industry.

KEYWORDS

artificial intelligence, deep learning, engineering drawings, engineering, procurement, and construction (EPC) industry, object recognition, piping and instrumentation diagrams (P&IDs)

1 | INTRODUCTION

Artificial intelligence (AI) is often perceived as today's most important general-purpose technology. AI has already triggered substantial changes in healthcare, manufacturing, transportation, retail, media, finance, oil, and gas and transformed their competition rules.[1–5] Various industries are currently considering applications of machine learning and deep learning (ML/DL) within their fields, applying image recognition, natural language processing, and data mining for operation optimization and knowledge discovery.[6] This trend also has been strongly supported by many government initiatives, like Industry 4.0 (Germany), Smart Factory (South Korea), and Smart Manufacturing (United States).[7] However, for introducing AI into the industrial sector, collecting, understanding, and using the massive amount of data generated by the Industrial Internet of Things (IIoT) is essential. The past decade has revealed AI's potential for inspection and maintenance applications.[1] The next step is to expand it into other application fields like engineering, procurement, and construction (EPC) sector and turning AI potential into their business values. This process will involve overcoming several challenges related to data quality, data pre-processing, and modeling and will include a variety of additional problems such as data collection, data security, or legal constraints. Those challenges are discussed in detail in the IEEE paper "Challenges in developing and deploying AI in the engineering, procurement and construction industry".[8]

1.1 | What is EPC

The EPC industry covers the entire industrial installation cycle, from bidding to engineering, construction, and start-up operation. The typical EPC project is a technology-intensive design that relies a lot on the experience of engineering and construction teams.

EPC projects utilize a contract-based project delivery model and are mainly applied for large-scale infrastructure projects in the private sector. These are prevalent in industries such as energy or oil and gas,[9] where companies often rely on EPC contractors for large-scale and long-term projects that require high-skilled labor due to the sector's complexity and high safety standards.[10] For example, as per IEC 61508 (an international standard published by the International Electrotechnical Commission), the minimum safety integrity level for the petrochemical industry allows only one dangerous failure in 100,000 h.[11]

EPC contract is often associated with a turnkey contract. The turnkey contractor holds all responsibility from the beginning of the project design until its start-up. The scope of work, in this case, includes the provision of engineering services, materials procurement, and construction services.[12] If anything goes wrong during the project execution, the EPC contractors must take care of the liabilities. Those contractors cannot go beyond a guaranteed price and must complete the project delivery before the fixed deadline. If the contractors fail to meet the deadline, they are predisposed to pay the delay liquidated damages (DLDs).[13,14]

The rapid development of the global economy, shown by the development of all industrial sectors, has increased the popularity of EPC projects. The need for EPC projects has been influenced by population growth, the nation's economic growth, and sustainable development concerns.[12] The main scope of each phase of the EPC contract is listed below:

- Engineering phase: basic engineering, detailed engineering, detailed design, and planning
- Procurement phase: logistics, transportation, purchasing, invoicing, and receiving the materials
- Construction phase: installation of electrical and mechanical equipment, construction management, and quality control and testing

In turnkey EPC projects, those phases are followed by commissioning and start-up.[13,15]

1.2 | Engineering—the core of EPC project

The engineering phase is, in fact, the preparation of plot plans, drawings, and specifications required for the project procurement and construction works. Engineering disciplines of energy, petrochemical, or oil and gas industries include process, process safety, mechanical, piping, electrical, control system and instrumentation, and civil, structural, and architectural engineering.[13] The engineering role in EPC project execution based on literature review[9,12–15] and McDermott's experience are summarized in Figure 1. It is visible that engineering is a core of EPC projects as it

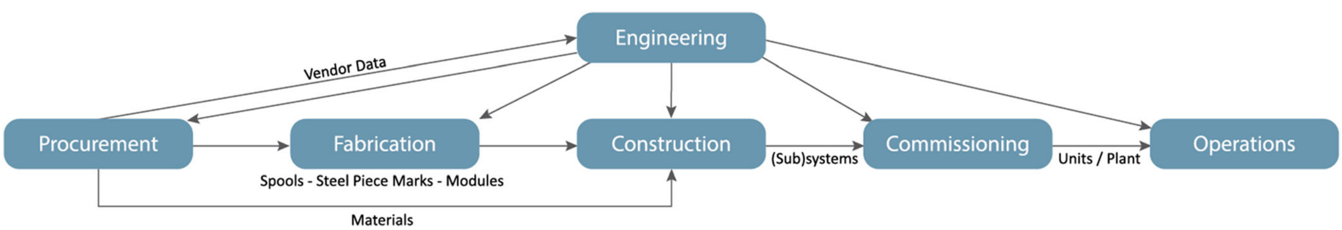

FIGURE 1 Engineering inputs during different project activities.[9,12–15]

provides input to every phase of the project execution. Procurement team buys nothing else than what engineering team specifies, construction or fabrication is done as per engineering drawings, and commissioning checks are done as per documentation issued also during the engineering phase.

With projects getting more lump-sum oriented, tighter in schedule, and work hours of specialists getting more costly, the demand for automating the repetitive work to release the highly skilled engineers for more complex tasks has become crucial for EPC companies to stay competitive. Therefore, in this research, we focus on AI applications that are supposed to reinforce the EPC project's engineering phase by improving quality and time effectiveness.

1.3 | Improving engineering drawings quality—potential AI application

One of the areas in which AI can have a noticeable impact is reducing the manual work of engineers working with complex drawings. Using the power of AI, it is possible to automate some of repetitive activities while keeping the performance on the level of a human expert. That can produce big dividends due to lowering the engineering hours and, therefore, cost, not mentioning the positive impacts on quality. In this research, we focus on the potential application, which is related to quality checks of piping and instrumentation diagrams (P&IDs) of large petrochemical installations. An example of such a drawing is shown in Figure 2. Until now, existing solutions have been limited to manual checks under the supervision of experienced engineers. Consequently, the quality check of a large number of P&IDs requires massive amounts of human labor, which could result in delays and cost overruns.[16]

1.4 | Research focus

Motivated by previous work done in symbol detection by DL algorithms, we combined it with McDermott's experience of EPC project execution in order to automate the manual check while keeping the same quality as the experienced engineers would do it. In other words, we combined the expertise of symbol detection on complex engineering drawings with engineers' practical experience to find the design mistakes that can cause significant delays in the construction and potentially jeopardize the overall engineering project schedule. To do so, McDermott AI team and domain Subject Matter Experts (SMEs) worked together to develop and train a DL algorithm based on the You Only Look Once (YOLO) neural network. YOLO is an open-sourced available framework that allows for simultaneous predictions of multiple bounding boxes and class probabilities using a single convolutional neural network (CNN). It is mainly used for a real-time object detection in video streams where the speed is essential because it can analyze tens of pictures in a second.[17] As a result, we created a software that could be potentially applied to other EPC companies doing similar design for complex installations with high safety standards like oil and gas or petrochemical plants because the design errors it captures are shared within this industry.

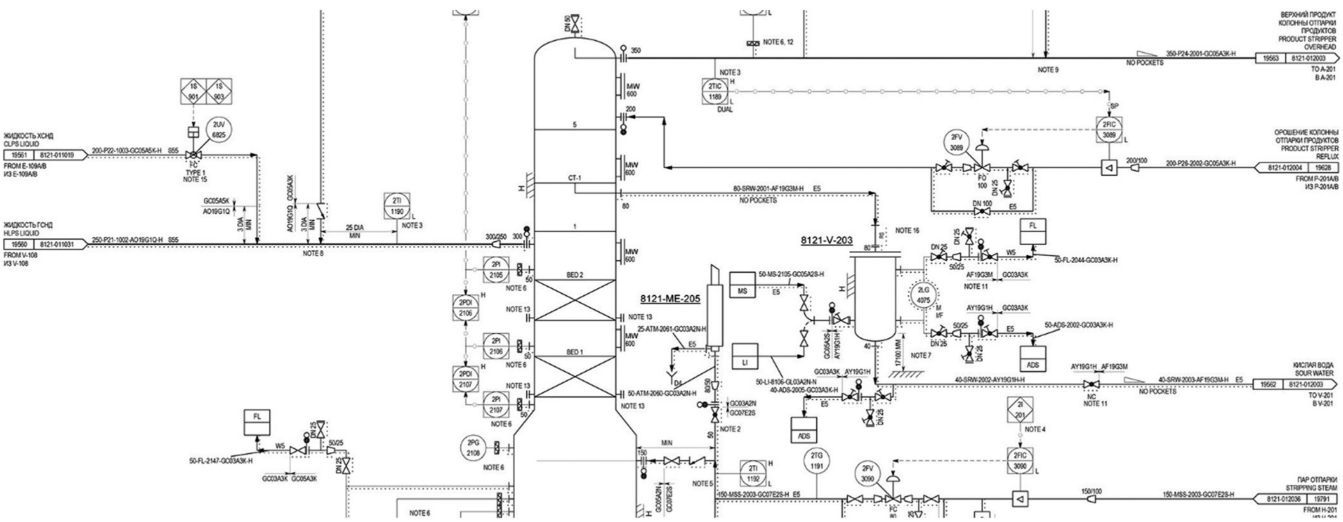

FIGURE 2 Example of a piping and instrumentation diagram (P&ID).

This paper is an extended version of a conference paper published at the IEEE World AI IoT Congress 2022. The published conference paper only focuses on the development of the DL model. This extended version includes the description of the final product development—AI-based software, details of its application, and cost-benefit analysis.

Our contribution is a proposed DL model and its training approach, description of the achieved results to date, cost-benefit analysis, and potential scale-up of the final product. It is worth mentioning that the developed DL model is domain agnostic. It can be re-trained on various schematic diagrams within engineering fields. Thus, the working principle could also guide other researchers to see whether similar solutions could be built for different industries. The model is trained and performs on noisy real-life data, which also can be essential feedback to the research community. Because the adoption of AI solutions is relatively low in EPC, particularly compared with other industries,[18] exploring AI-powered applications is essential for this industry. Therefore, another goal of this paper is to share knowledge of real-life application development to enrich the AI experience in the EPC field.

The paper is organized as follows. Section 2 is a section with related work. Section 3 describes the research method, including the research process description and data collection. Section 4 discusses the problem identification and solution development, including its evaluation. Performance analysis is described in Section 5. Section 6 discusses how these results improve current practices. Threats to validity and our conclusion are presented in Sections 7 and 8.

2 | RELATED WORK

2.1 | AI development in engineering, procurement and construction (EPC) within the energy sector

Many companies and experts believe that AI can accelerate digitalization and improve the application of data-driven intelligence systems for practical engineering in the future.[19] In recent years, researchers have published articles on applying AI to tackle EPC-specific challenges in the energy and petrochemical industries. For instance, machine learning has been used for cost estimation of engineering services,[20] health and safety monitoring, specific material cost estimates, supply chain, logistics process improvements, and risk prediction.[21,22] Robotics has been applied in the inspection and maintenance sector for surveillance systems and optimizing maintenance plans.[1,23] Knowledge-based systems have also been used for tender evaluation, conflict resolution, risk and waste management, sustainability assessments, and so forth.[22,24] Although, there are still difficulties that exist with data exploration to let ML/DL support these industries. In the work of Koroteev and Tekic,[1] the authors reviewed the energy sector and described the learnings from AI projects for leading oil and gas upstream companies worldwide. Reviewed research identifies that existing data collection and storage solutions, like poor logging and limited data cleaning mechanisms, are insufficient to set up ML/DL systems. Alongside the data gathering and storage process challenges, they also highlighted challenges related to the people and their knowledge of AI. Although oil and gas operators like BP, Shell, and Saudi Aramco are investing aggressively in AI startups and R&D, those challenges prevent them from implementing AI on a large scale.

2.2 | AI for symbol recognition and digitalizing engineering drawings

Many EPC companies have adopted model-based solutions like *AVEVA Engineering* or *SmartPlant* provided by AVEVA (aveva.com/en/products/aveva-engineering) and Hexagon (hexagon.com/solutions), shifting the industry design process to 3D modeling. However, even nowadays, engineers might still receive drawings as generated PDFs or scanned documents. Sometimes, the exchange of engineering design between EPC contractors and their clients is done in PDF based on pre-agreed arrangements. Therefore, several scientific studies and practitioner-oriented white papers reveal potential opportunities in digitalizing engineering drawings while many enterprises either start partnering or outsourcing their AI projects to thousands of new AI start-ups.[25]

Although digitalization of engineering drawings with the help of AI still requires more attention in the research community,[26] several works on this subject have been recently published to help practitioners.[26–31] Since 2010, some researchers have begun to use neural networks and machine-learning algorithms for pattern recognition, recognition of text, lines, and symbols from scanned images or PDFs, and automatic generation of digitalized drawings.[32] The process involved identifying various text codes and symbols and localizing them in a very complex structure. Extensive studies on the latest trends in the digitalization of complex engineering drawings have been done by Moreno-García et al.,[27] where authors provide an overview of what DL can achieve based on supervised learning. In particular, they highlight the application of CNNs. CNN-based models, in general, are preferred models for symbol classification among others due to their capability to classify a vast pool of images of various sizes and characteristics with high accuracy.[27,33,34] In the work of Nurminen et al.,[17] authors particularly studied the YOLO neural network algorithm to detect high-level objects, for example, pumps or valves, in drawings scanned from hard copy versions. So far, no references are found, where these techniques are applied to spot the engineering errors on drawings made for complex installations like oil and gas or petrochemical plants, where besides the conventional problem of variability in size, direction, position, or image quality of symbols, developers have to deal with various symbol representation due to different client standards.

3 | METHOD AND DATA COLLECTION

3.1 | Case company

This research has been performed at McDermott—a large EPC company operating in over 54 countries and includes approximately 40,000 employees. The company operates a diversified fleet of marine construction vessels and fabrication facilities and performs oil and gas, petrochemicals, and energy transition projects design and construction worldwide. The research was performed by an AI team that was gathered from the company's engineering forces. The team also involved the domain experts from the process department of The Hague office of McDermott.

3.2 | Research process

This work can be identified as action research because the first two authors are members of the AI team and have been actively involved in the final solution development described in the next section. The authors' involvement in the product development allowed simultaneously study the experience while solving the real-life engineering problem. Although the role of a research collaborator cannot be seen as an objective reporter,[35] it provided rich insights into the development processes because the authors were involved in the day-to-day happenings. That would not have been possible for an outside observer due to the confidentiality of data used during the research. The details about the research team members and their roles within the project are presented in Table 1. The first two authors of this paper are indicated in brackets.

To analyze the empirical results, we structure them into a particular set of activities, which were conducted using an iterative approach. These activities are derived from action research workflows available in academia and used among practitioners (Figure 3).[36–38]

In all, 12 people were involved in the solution's prototype development, which took place between January 2021 and January 2022 (Figure 4). The prototype included the DL model and simple user interface to allow domain experts to do the performance evaluation. The performed activities during that period did not include the software development.

TABLE 1 Research team details.

Job ID within organization	Role in the research
Senior engineering manager	Project manager (first author)
Principle process engineer	Domain experts
Junior process engineer	Domain expert and data scientist
Instrumentation and control system engineer	ML model developer and technical lead (second author)
Junior instrumentation and control system engineer	ML programmer/front-end developer and data scientist
2 x junior instrumentation and control system designers	Labelers
9 x junior process designers	Labelers
10 x principle and senior process engineers	Evaluators

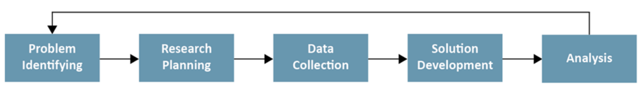

FIGURE 3 Typical performed activities for action research.[36–38]

FIGURE 4 Timeline of performed activities.

3.2.1 | Problem identifying

The main challenge for the research team was to find a solution that would reduce a manual work of engineers. It was decided to investigate how AI can help in automating some of the repetitive work within the process discipline. It was essential to involve the future product users, that is, process engineers. Process engineers were asked to identify the unique domain challenges that still require experts' intervention. After screening a typical EPC project workflow, engineers identified the steps where manual work is heavily involved. One of those steps is a manual quality check of the engineering design. Although many EPC companies use 3D modeling, not all design mistakes can be revealed there. A manual review is still required when 2D drawings are generated from the model. It is also applicable when engineering drawings are received from the licensor or client in 2D format and need to be reviewed. This review usually takes hours and requires an experience to timely spot design errors that can jeopardize the construction phase of the EPC project.

To see whether ML/DL technology could provide a solution, the AI team worked extensively with domain experts to study the quality review process. It was decided to focus on the engineering errors on the P&IDs that cannot be found by any existing software and, therefore, are required a manual review. The next step was to find a way on how to learn the model to recognize those errors through the supervised learning. To do so, researchers identified the patterns that represent engineering mistakes and their correct arrangements to create extensive training datasets.

3.2.2 | Research planning

After studying the domain problem, the research team identified what type of ML/DL model might help and what is required regarding data and resources. The desired model output was the highlighted patterns with a "correct" or "incorrect" label attached to them, so engineers could easily find those labels and correct the design if required. To achieve pattern recognition, the object recognition technique was applied. The task was divided into two parts: first—object localization, that is, classifying the object's (pattern) location and drawing a box around it and second—object (pattern) classification and assigning the predicted class (correct or incorrect representation). An initial execution plan, budget, schedule, and critical project milestones together with the appropriate success metrics were created. Success metrics were defined to measure the business value, which the future solution might bring. For this particular case, the success criterion was the saved hours of engineers.

3.2.3 | Collecting the data

Research data were collected from more than a dozen large, multi-billion EPC projects executed by McDermott during the last 15 years. Data were collected by engineers who also preprocessed and prepared the training/test datasets. The details of the research team can be found in Table 1.

3.2.4 | Solution development

While developing a solution, we focused on the following research question: "How to automate a manual check of the P&IDs with the same or higher quality as a human expert?"

We based our research question on pre-existing knowledge of deep learning algorithms and their application for symbol recognition on complex blueprints. Thus, we used the solutions known to academia and applied them to a specific real-life problem. The idea of engineering error recognition by deep learning algorithms can be applied to various fields. Therefore, the learning outcomes of this research might be helpful to other practitioners and academia.

3.2.5 | Analysis

In this phase, the AI team involved domain experts to check the model's performance on new data. The assigned experts were from different offices of McDermott: The Hague (the Netherland), Brno (Czech Republic), and Gurgaon (India). In Table 1, they are indicated as evaluators. The final analysis also included the cost comparison between investment in development and money value of saved engineering hours.

4 | SOLUTION DEVELOPMENT

Below, we detail the solution development using the action research phases as per the structure described in Section 3. In Section 4.1, we describe the examples of mistakes, which the developed model can spot. These mistakes are not unique to McDermott's design and can be found in the design done by any other EPC contractor operating in the energy sector. In Section 4.2, we describe the data collection process and challenges that the developing team faced while preparing the labeled dataset for model training, that is, how the class imbalance is solved. Section 4.3 focuses on the model training and performance evaluation and shows the example of its output.

4.1 | Examples of engineering mistakes on P&IDs

4.1.1 | Drain valves in control valves with fail close (FC)

Based on specific licensor requirements, control valves with FC mode need a bleed/vent before and after the control valve. Double vent or bleed valves become necessary after shutting down a section of pipe where a control valve failed with "close" action. In this way, both sides can be drained and/or release its pressure safely. It is required in case of emergency or planned maintenance of the control valve (Figure 5).

Design mistake frequency: medium-high.

Cost impact: medium-low. In this case, a new vent valve/bleed is required to be installed. The cost will rise if the mistake is found only during construction because the installation might cause a delay in other construction work and impact the overall schedule.

4.1.2 | Check valves upside down

The check valve shall be installed such that it can regulate the flow of medium in the pipes. Usually, there is a direction mark or dot on the valve body. If installed upside down, most check valves will not close properly, producing a back flow event, which can damage the system that the valve was installed to protect (Figure 6).

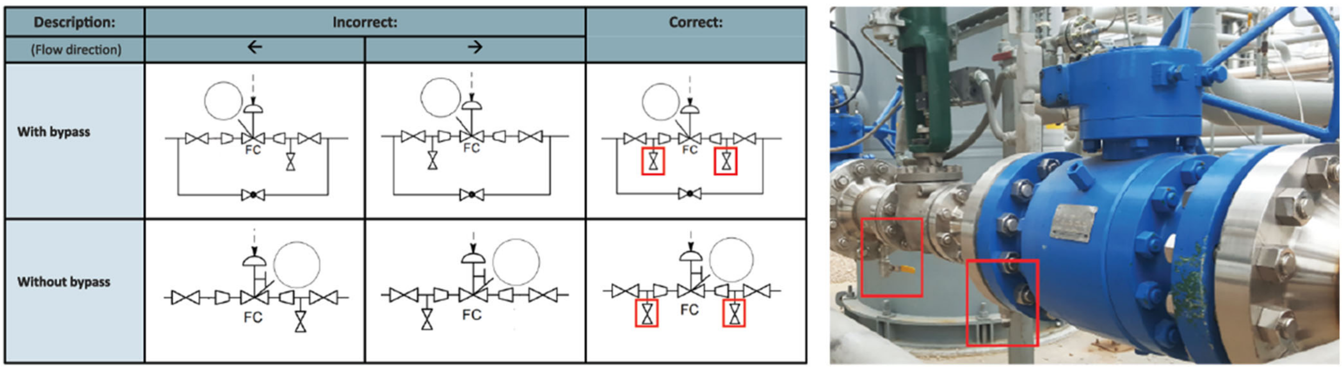

FIGURE 5 Graphic example of control valve with double bleed—patterns of correct/incorrect installation.

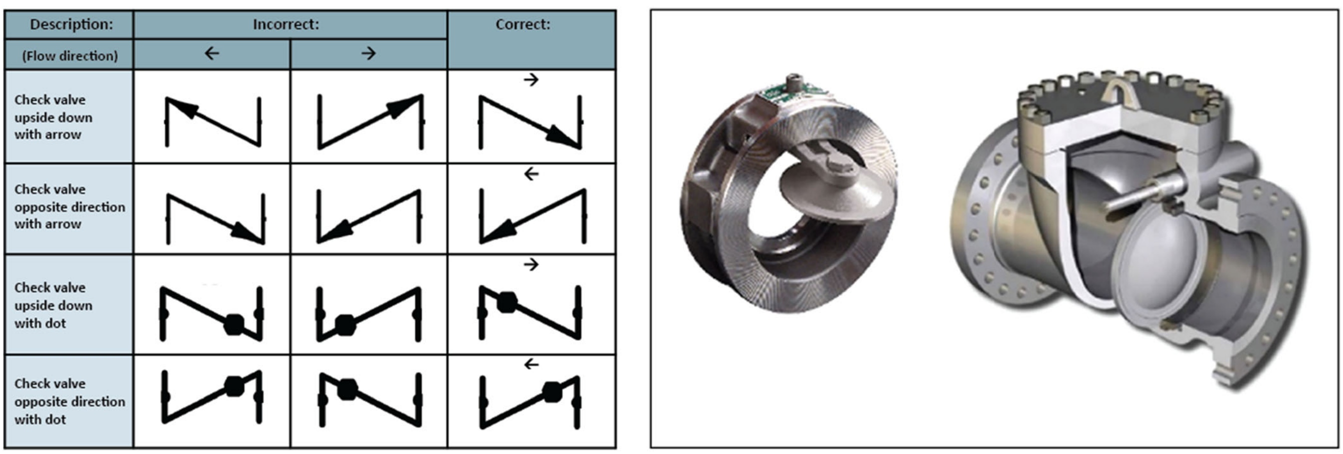

FIGURE 6 Example of check valve—patterns with correct/incorrect direction identification.

Design mistake frequency: upside down—medium. Mainly lines that were drafted vertically.

Cost impact: upside down—medium. It is easy to modify the orientation depending on the check valve size.

Opposite direction—high. The impact of a check valve in the opposite direction flow can cause severe damage to equipment and piping lines.

4.1.3 | Pump warm-up bypass. Globe valve direction

The globe valve bypass aims to warm up the pump line. This warm-up bypass is required when the plant starts up or when a spared pump is on standby mode and ready to replace its twin. This bypass around the check valve is only required for certain process conditions (Figure 7). If the globe valve bypass is not in the correct direction, the valve can be broken, and the pump will not be warmed-up, which will cause pump damage.

Design mistake frequency: medium. It is lightly common to forget the arrow in P&IDs.

Cost impact: medium. Not being able to warm up the system can damage the pump.

4.1.4 | Butterfly valve with blind

The butterfly valves need additional space in the line to be able to swing properly. Thus, the isolation elements, such as the blind, need to be at some distance from them (Figure 8).

Design mistake frequency: medium. Butterfly valves are only used for lines of big sizes.

Cost impact: medium. This mistake requires a spool (a pipe) to separate the valve and the blind.

4.2 | Data collection

Dataset preparation is the most crucial step for any machine learning application. The quantity, as well as the quality of data, heavily influences machine learning model accuracy and precision.[39] The straightforward application of CNNs for the digitization and contextualization of complex

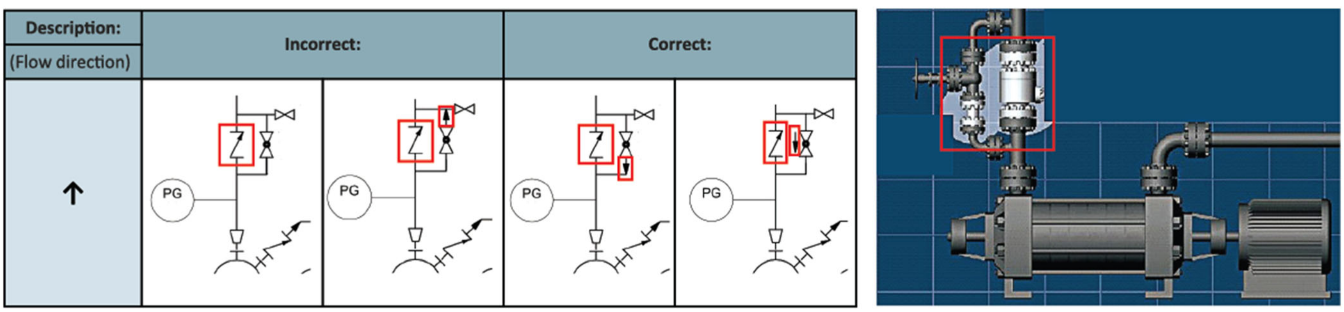

FIGURE 7 Pump warm-up bypass—patterns with its correct/incorrect globe valve directions. PG, gauge pressure.

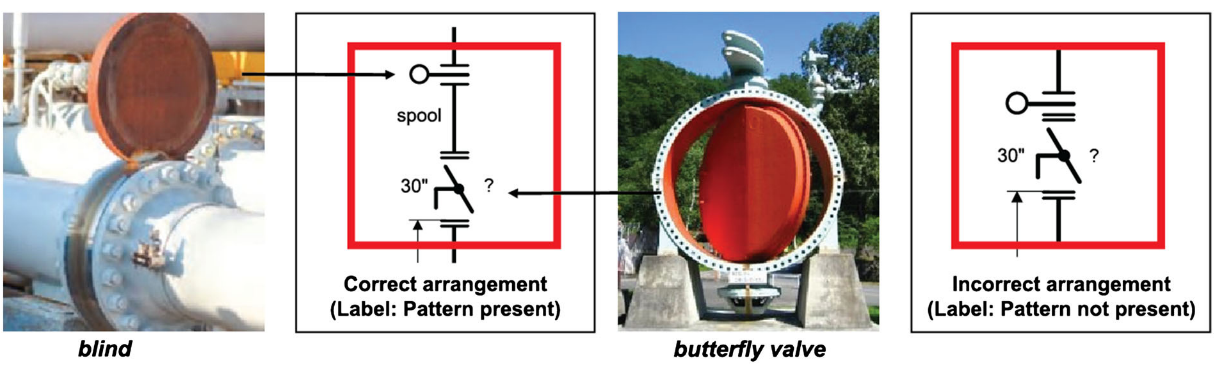

FIGURE 8 Butterfly valve and a positive isolation element. Example of correct and incorrect installation.

engineering drawings is still challenging due to the lack of sufficient annotated examples in industrial practice. While some general-purpose symbol repositories can be found in literature, there are no domain datasets for diagrams such as P&IDs where symbols of different standards are collected. In addition to the typical computer vision problems such as light, scale, and position variations, symbols on P&IDs can follow different standards, that is, the client's internal, specific countries, or international standards.[27]

Therefore, compiling a well-defined and clearly labeled dataset that can be used for symbol classification is a crucial task where domain experts shall be involved. Many clients used in the past, and some of them at the current time, their own standards for representing items on the drawings. That leads to extensive variations of each symbol. In addition, the text characters may overlap with symbols or other characters. The data collection, in this case, requires lots of various projects screening, which are developed for different clients. However, the chance that new clients would use their own symbology is still high. For example, the variations of symbols in the pattern on Figure 8 were above 200.

4.2.1 | Gathering raw data

Large EPC companies work worldwide, and projects are being executed from offices on various continents. Even though there are central databases available where data files are saved for future references, these drawings are not shared consistently, making it challenging to use for AI-based solution development.[29] This is also applicable to McDermott. Therefore, the first step in this research was to gather P&IDs from as many projects as possible. Because the deep learning algorithm was based on object detection, a lot of images were required to achieve acceptable accuracy.[40] With the help of other departments, the research team gathered over 15,000 P&IDs from 15 large multi-billion EPC projects executed within McDermott.

4.2.2 | Data pre-processing—addressing a class imbalance

The DL algorithm must identify and classify engineering mistakes as false representations of the symbols' assembly (incorrect pattern) from the correct representation of the same assembly (correct pattern). To avoid class imbalance, the dataset needs to have a sufficient, preferably equal amount of all the variations of identified patterns in their correct and incorrect forms.[41] However, during the initial data analysis, it was discovered that some patterns occurred only a few times throughout the training set of P&IDs. To tackle a class imbalance, different variations were created by changing the orientation or text/symbols associated with each pattern. The biggest challenge was to ensure that the augmentation process produced results as close as possible to real-life variations of the correct/incorrect symbols' assembly. For example, in the control valve case (Figure 9), "FC" indicates that the valve is supposed to close in case of any process failure, and the dotted line is a symbol of electrical heat tracing around the pipeline and valves. Every text and symbol has a specific purpose/context behind of why and where they are placed, which depends on the process technology represented on drawings and, therefore, might vary from project to project. Thus, domain knowledge is required to create realistically augmented patterns.

4.2.3 | Labeling the dataset

Our practice showed that the high-quality labeled training sets might drastically improve the model's performance. That as well has been confirmed in the literature. According to Northcutt Curtis et al.,[42] low-capacity models with high-quality training test set in noisy real-world

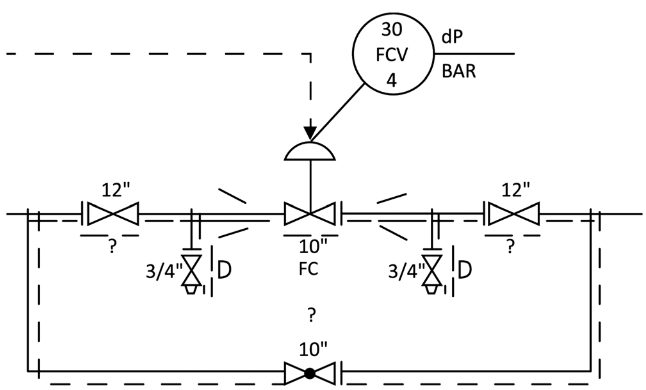

FIGURE 9 Graphic example of a control valve with fail close (FC). BAR, a metric unit of pressure; dP, differential pressure; FCV, Flow control valve.

applications may outperform high-capacity models. Labeling, in our case, was a challenging activity because each item on the drawing had a tag number or numeric value, indicating specific parameters like the angle of the valve position, for example.

Sometimes, those values are too close to the symbol and require very accurate labeling to eliminate numeric values or tag numbers within the label boundaries. In addition to this, the texts or numbers around symbols could be crucial for correct or incorrect pattern representation. Therefore, to avoid "Garbage in–Garbage out," which makes it difficult to achieve acceptable model accuracy,[43] the research team had to ensure that in such congested drawings like P&IDs, no irrelevant symbols or text would be captured inside the labeling bounding box. Due to these complexities, it was decided to involve again the engineers and designers with domain knowledge who were aware of the context of different symbols and texts on the process lines. They manually labeled the dataset and included all pattern variations. Because multiple people were doing the labeling, it was essential to ensure uniformity across all the labeled files. For example, all the labels should have the right names, spellings, and class IDs of labels. Because of the sensitivity of the data, it was essential to ensure that it was not uploaded onto any third-party platform. Therefore, a labeling tool was required to be installed on local machines. The research team selected LabelImg windows version 1.8.0 (Figure 10). It is an open-source python-based image annotation tool written in Python that tags the object's class and position information and records them as XML files in PASCAL VOC, YOLO and CreateML formats. The PDF drawings were converted into images for labeling before using this software. It was also required to generate the annotation .txt file for the developed YOLO-based DL model. Thus, the final training dataset consisted of annotated data (.txt files) containing class IDs and coordinates of the bounding boxes' center, width, and height. The training dataset consisted of 2253 images, including 8392 annotated patterns of eight different classes.

4.3 | Model development

4.3.1 | Selecting model type and architecture

With the rapid growth in AI field, many algorithms and models are available for deep learning applications. After studying various DL model types, the research team decided to proceed with object detection type. Object detection models are easily trained to identify and locate any number of symbols/patterns in the P&IDs. During the R&D phase, the model was hosted, trained, and tested on Google Colaboratory (pro+), a cloud-based service from Google for developing machine learning applications, which also allows code sharing.[44]

After finalizing the model type selection, the next step for the research team was to decide on a network architecture for implementing object detection on P&ID images. The original image's resolution size of 6000 × 4000 had to be scaled down to 1024 × 1024 due to the limited processing capabilities available. Reduction of the resolution size, though, posed a big challenge as the patterns were very small compared to the overall image size. During the initial attempts to resolve this challenge, the research team prepared the dataset with an increased size of a single trial pattern by manually copy-pasting the pattern with a larger size than the rest of the P&ID image (Figure 11).

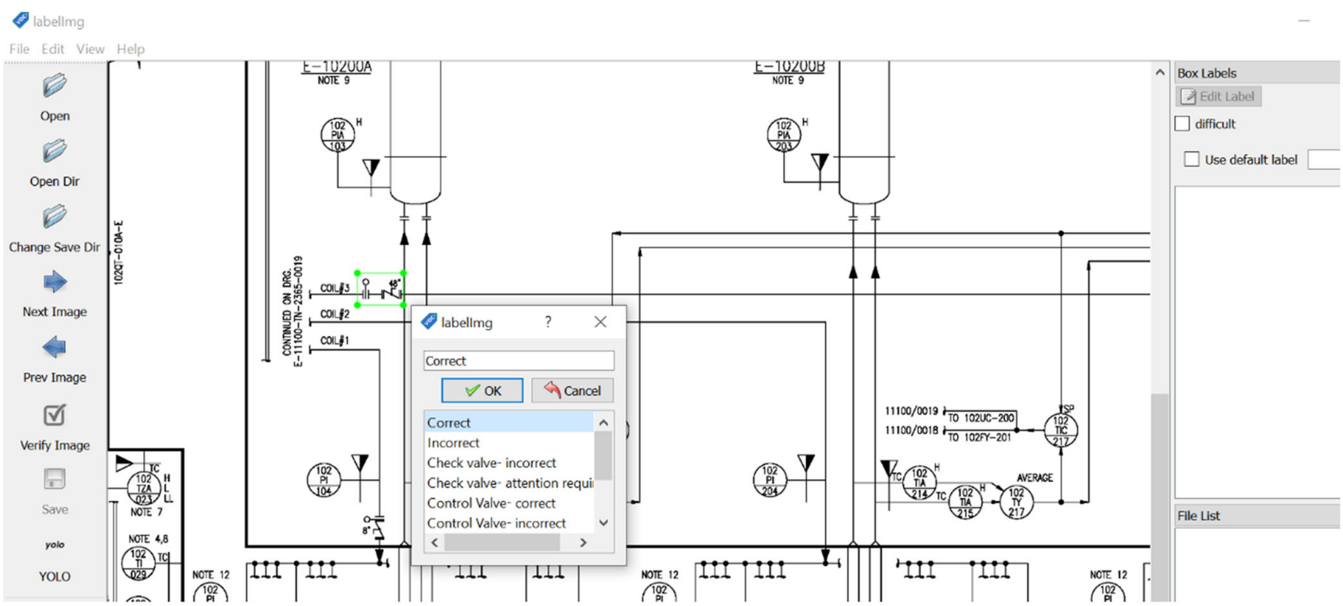

FIGURE 10 Example of labeling process in LabelImg.

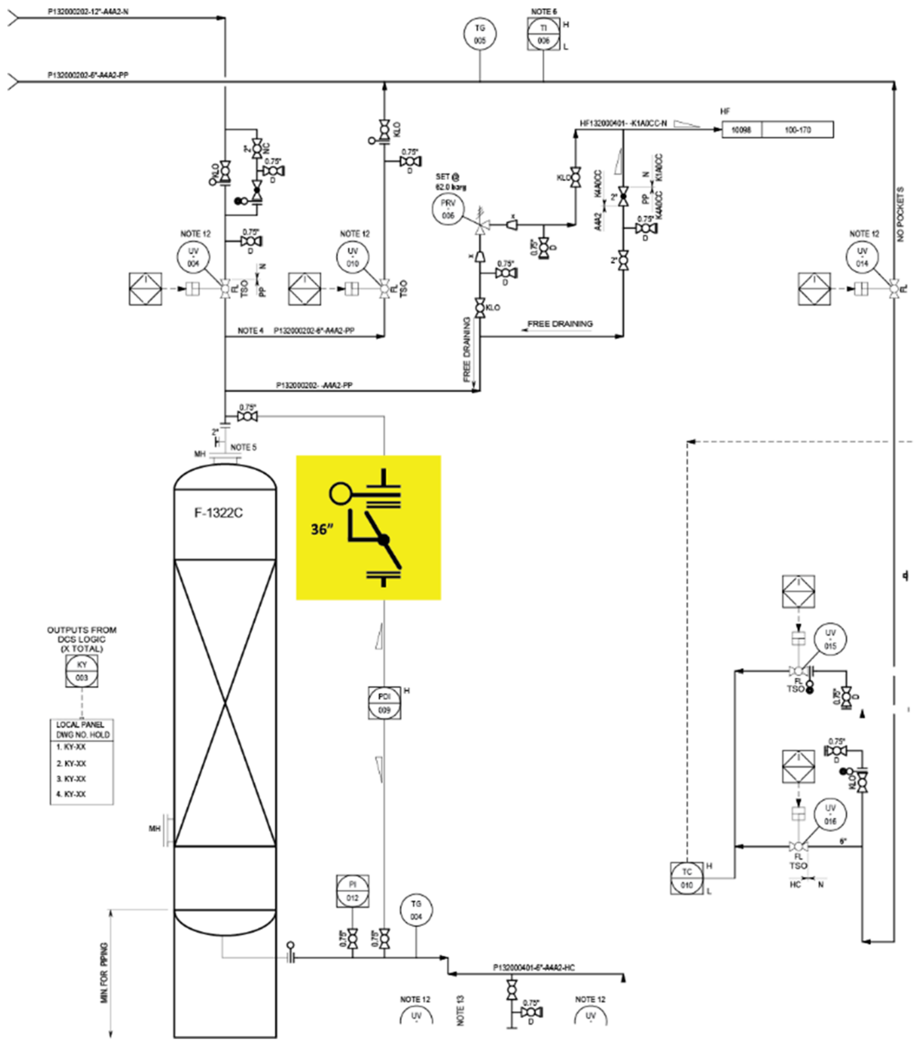

FIGURE 11 Piping and instrumentation diagram (P&ID) with manually increased size of the pattern.

At first, a MobileNetSSD object detection model was used on the manually modified dataset. However, the model failed to detect pattern completely. This outcome reconfirmed the statement mentioned by the model developer that single shot dedector (SSD) does not work well for small object detection.[45] The next model that the team tried was the EfficientDet (D0) TensorFlow 2. This model was able to detect the pattern with increased size (Figure 11). However, when it was tested on the new non-modified dataset, it also failed to detect the same pattern. Therefore, the research team decided to discard the manually modified training dataset, concluding that using the artificially increased pattern size misled the model because it only started to search for the patterns with increased size and ignored the original resolution of the pattern on the drawings. Lastly, YOLO v4 was tested, and the model could detect the pattern with the original size on the images of 1024 $\times$ 1024 resolution. Therefore, the research team decided to continue with the YOLO v4 model and successfully expanded it to detect other patterns too. There are additional advantages of the YOLO v4 model that also supported its selection, that is:

- It has a simple architecture based on a single convolutional network that can simultaneously predict multiple bounding boxes (objects) and class probabilities.
- As stated by its developer by Bochkovskiy et al.,[46] YOLO v4 was created to have an efficient and powerful object detection model that uses a single graphics processing unit (GPU) during training, making it a high-speed and accurate object detector. This fact significantly helped the team to deal with processing power limitations.
- YOLO sees the entire image while analyzing. Thus, it implicitly encodes contextual information about classes and their appearances.[47] In this research case, the location and context of these assemblies were necessary for the end-users.
- The components in the selected patterns were very small compared to the entire image. YOLO v4 performed well in detecting these small-sized components. Srivastava et al.[48] also confirm similar findings.

- Data preparation for YOLO v4 is fast and easy,[48] which helps to include a significant number of patterns and images. The research team was able to take help for data annotation tasks from domain experts with non-IT or programming backgrounds.

4.3.2 | Training

YOLO v4 architecture was customized to the number of classes applicable in this research. These classes represented different pattern types the model was trained to recognize. At the moment of writing this paper, the model had been successfully trained and tested for eight classes (eight pattern types representing four typical design errors and their correct versions). The number of filters was modified in each of the three YOLO layers according to Formula (1):

$$\#\ of\ filters = (\#\ of\ classes + 5) \times 3. \tag{1}$$

The input for the model was the set of P&IDs in PDF format converted to PNG files with resolution 6623 × 4678, which were resized to 1024 × 1024 for training, and the .txt annotated files generated during the labelling process. Around 4506 files (2253 images and 2253 .txt files) were used as the dataset, which was split into 90%/10% as training/test datasets. The network, which used the Mish activation function,[41] was trained with a learning rate of 0.001 with the max number of batches being 16,000, where each batch size is equal to 64 training examples, and 12,800, and 14,400 steps.

4.3.3 | Results

A test dataset was gathered from P&IDs received from a new client of McDermott. Therefore, the symbology of the patterns varied from the symbology used in the training dataset. Nevertheless, the model recognized around 70% (160 patterns out of 230), with a confidence percentage above 30%. After retraining the DL model on new symbology variations, the performance improved to almost 100% (309 out of 310) with a confidence percentage above 50% for all predictions. The majority of class IDs were accurately detected and recognized. In other words, the patterns, which have enough examples of different symbols' variations in the training set, were accurately identified in the test dataset.

It is essential to underline that the symbols' sizes significantly influenced the detection. When the model scaled down the original image resolution for analysis, the small symbols lost many of their characteristic features and became so blurry that classifying them correctly was difficult even for a human eye, not to mention the model. The same is confirmed in the research presented by Nurminen et al.[17] In our case, the original resolution was scaled down to 1024 × 1024. Training the model on this resolution in the Google Colaboratory (Pro+) required 7 days.

The training achieved a mean average precision of 80.6% with an IoU (Intersection over Union) threshold of 0.5 (mAP @ 0.50). The performance parameters obtained on the validation set of 234 P&IDs are shown in Table 2. These parameters were calculated by comparing the model predictions with the ground truth for these images. The model could predict 684 boxes correctly, represented by the TP (true positive) value in Table 2. The average IoU, which represents the overlap percentage between the predicted and ground truth bounding box, is 61.76%. FN in the table represents false negative, that is, a number of patterns the model has missed. For end-users, in this case, each FN means a missed design error. The tests results showed a high Recall meaning that FN is low versus the total number of detected patterns (Equation 2). There are ways to improve it further, for example, retraining the model with a lower confidence threshold value or training on improved image quality, which requires higher processing power.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Meanwhile, high FP (false positive) is not that crucial because engineers are supposed to review the model output and decide whether to accept the highlighted pattern as an engineering error. Figure 12 shows how the model could detect the relevant component in a complex real-life P&ID. It could find the essential features and was not misled by the incoming connections or overlapping text.

TABLE 2 Performance parameters.

Precision	Recall	F1-score	TP	FP	FN	Average IoU
0.82	0.84	0.83	684	146	130	61.76%

Abbreviations: Average IoU, average Intersection over Union; FP, false positive; FN, false negative; TP, true positive.

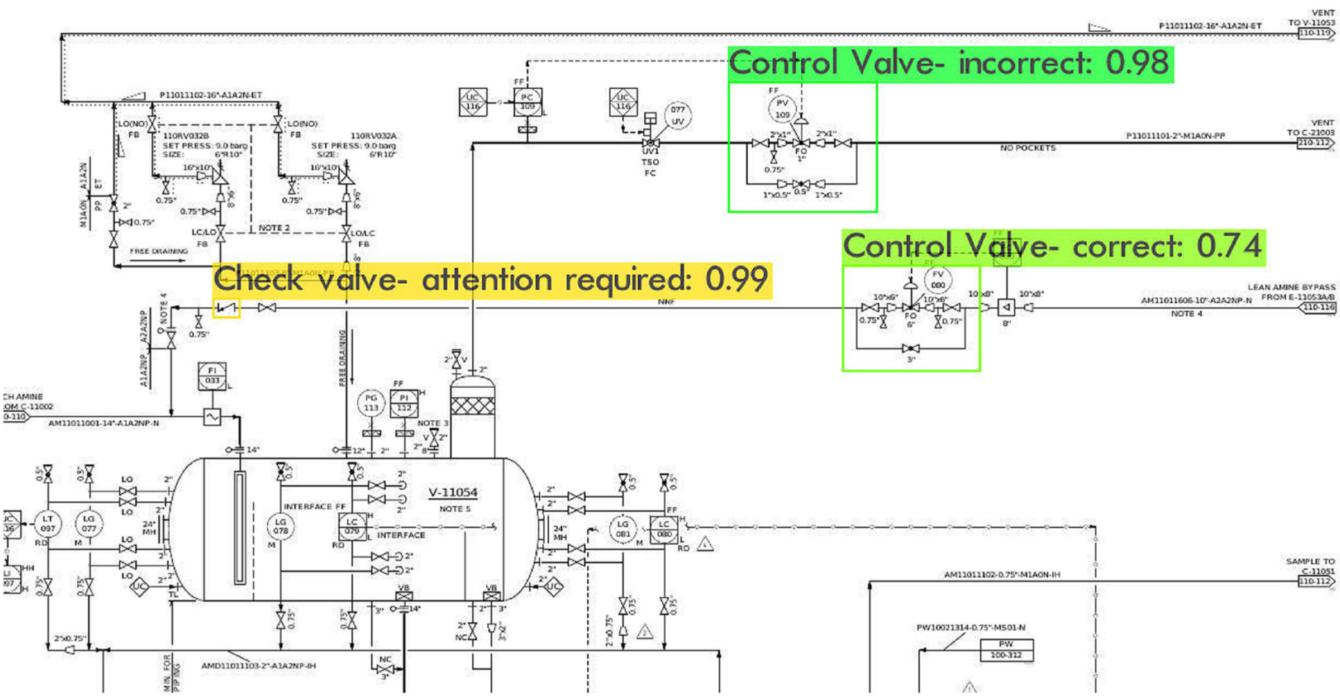

FIGURE 12 Detection of the design mistakes. Model's output example.

It is essential to note that the evaluation team also included domain experts—McDermott's process engineers. Even though Google Collab is an excellent platform for developers, research team experienced that it was difficult for people without programming backgrounds to use Google Collab notebooks. Therefore, domain experts required a simple user interface to evaluate the model. At first, the research team ran the model on local machines directly and discovered that running the application on the local machine was a resource-heavy exercise for the average computer and, therefore, took considerable time (several hours). Thus, for the evaluation phase, a simple collab-based user interface was developed to allow domain experts to test the model for a set of PDFs in one go in a cloud. Due to memory limitations, only a collection of 30 PDFs could be processed at a time. Because the drawings in PDF format had to be converted to images with a resolution of 1024×1024, the output images were of poor quality. Domain experts had to use output images only as a reference and search those identified mistakes on the original P&IDs manually. Those two major drawbacks—limitations on the PDF batch size and poor output image quality—forced the AI team to search for another solution before the deployment phase, that is, to develop a software that can host and operate the model.

4.3.4 | Software development

As well described by Sculley et al.,[49] only a tiny fraction of real-world ML systems is composed of the ML code itself. Usually, it requires a surrounding infrastructure for life cycle management that includes hosting the model, retraining, model adjustment, and provision for a data security. To verify the smoothness of ML code integration with target software, the pilot deployment to only a limited number of users was applied. It was essential for the research team to run user acceptance tests to handle possible issues before a full-scale rollout.

The developed AI tool for detecting design mistakes on P&IDs could be referred to ML operations (MLOps) of level 0—where building and deploying ML models is an entirely manual process (Figure 13).

In our case, MLOps include the following activities:

- Actively monitoring model during deployment
- Retraining for new engineering mistakes
- Continuously experimenting with data and model to improve the performance

In order to do this, the software shall be able to host the model where it can be retrained and redeployed. The primary requirements for the software in this case are

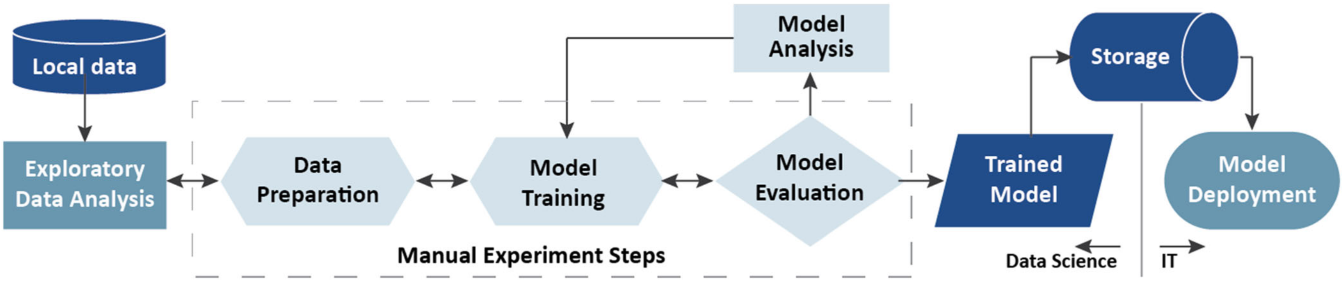

FIGURE 13 The workflow of ML operations (MLOps) level adopted from Google (https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning).

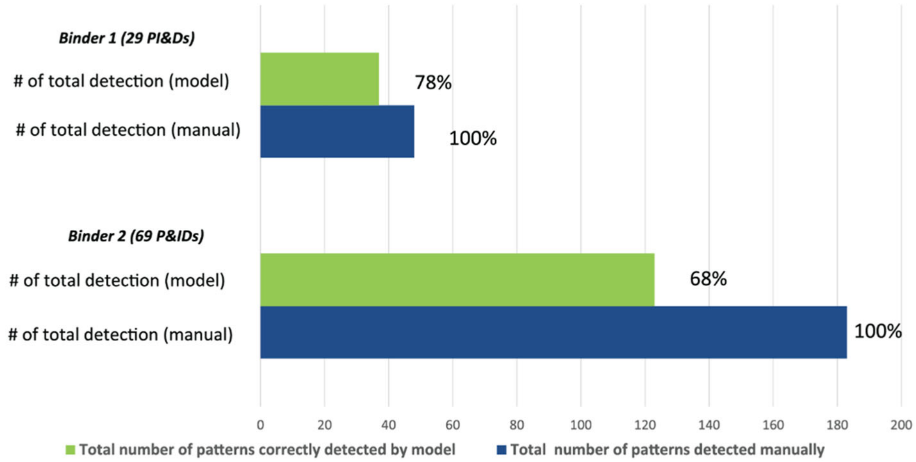

FIGURE 14 Manual check versus automatic check by deep learning (DL) model performed on the evaluation set containing new variations of symbology. P&IDs, piping and instrumentation diagrams.

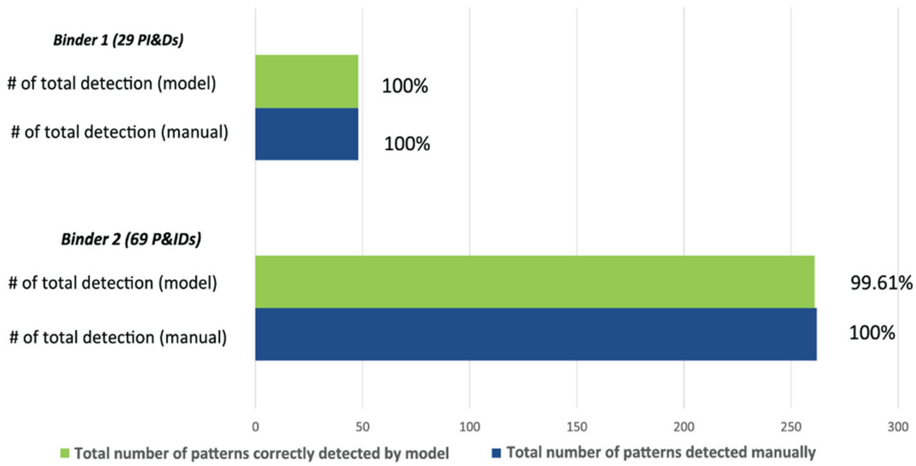

FIGURE 15 Manual check versus automatic check by deep learning (DL) model performed after retraining on new symbology variations. P&IDs, piping and instrumentation diagrams.

- Data and model security
- Intellectual property (IP) protection
- User interface for ML programmers and data scientists for model management
- User interface for process engineers—users of the tool

Two options were considered for software development: either by the company's IT department or an external party. In this case, the decision had been made to partner with an AI platform provider whose infrastructure satisfied all McDermott's requirements to host the developed DL model. It is worth mentioning that both options assumed the additional non-foreseen cost for external support or for hiring new staff personnel in IT departments.

5 | PERFORMANCE ANALYSIS

To decide whether or not the developed solution could achieve the research question, that is, whether it can help to reduce a reviewing time of P&IDs, the model's output was compared versus manual check by process engineers (Figures 14 and 15).

A dedicated team of process experts was gathered to estimate the time to be spent on manual reviews of the same set of drawings and evaluate the quality of the model performance. The evaluation set was gathered from P&IDs received from a new client of McDermott. Thus, the symbology within the patterns varied from the symbology used in the training dataset. Nevertheless, the model recognized around 70% (Figure 14). After retraining the DL model on new symbology variations, the performance improved to almost 100% (Figure 15). That means the symbology variations in the training set significantly influence the model performance, proving that a high-quality training set containing as many symbol variations as possible is crucial.

5.1 | Cost analysis

During the evaluation, process engineers spent around 6 h going through two binders of 98 P&IDs in total. Therefore, 6 h per 100 drawings (rounded up number) could be claimed as a saved time. Assuming that the check shall be done at each new version of P&IDs, it is possible to

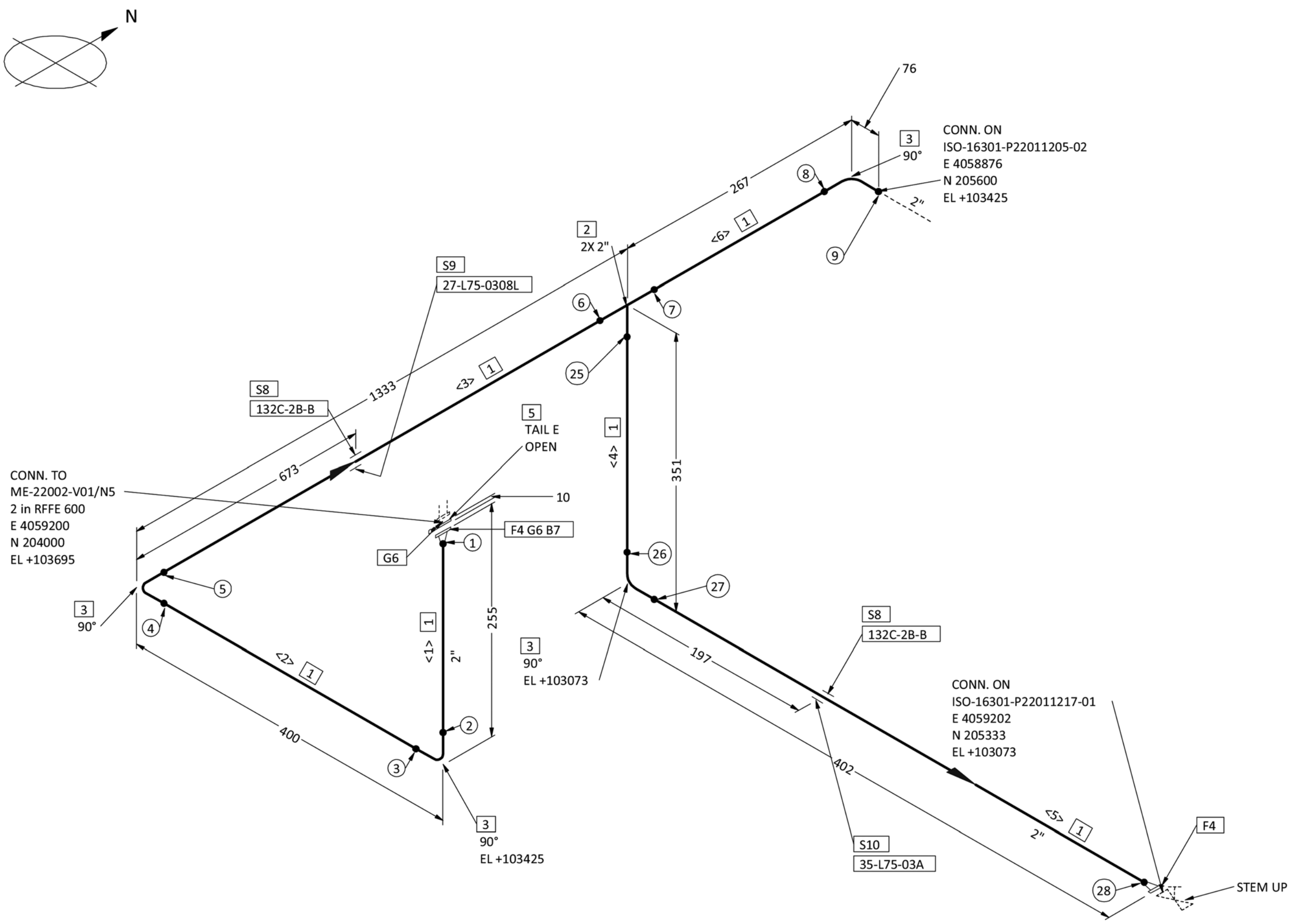

FIGURE 16 Example of piping diagram (Isometric).

estimate the total saved cost. During large EPC project execution, a new version of the same drawing can be issued in every single office each week. The average project within McDermott can contain a set of 200 up to 1000 P&IDs that need to be updated during each new revision. On average, a single engineering office of McDermott maintains five projects per year. It is realistic to assume that the cost per engineering hour is \$100 in high-cost and \$50 in low-cost countries. This amount includes the salary and an overhead cost, for example, equipment usage, office utilities and rent, taxes, insurance, and pension that the company pays for each employee apart from the salary.

With these assumptions, it is easy to calculate the monetary value of the saved time for engineers. The calculation of the potential cost benefits of the company is presented below:

- Number of P&ID to be checked per year in a single office: 600 drw $\times$ 52 weeks $\times$ 5 projects/year $=$ 156,000 drw
- Total saved hours per year in a single office: 6 h $\times$ 156,000 drw/100 $=$ 9360 h/year
- Cost saving in high-cost engineering office: 9360 h/year $\times$ 100 \$ $=$ 936,000 \$/year
- Cost saving in low-cost engineering office: 9360 h/year $\times$ 50 \$ $=$ 468,000 \$/year

McDermott has large engineering offices in high-cost countries like UAE, United Kingdom, United States, and the Netherland, as well as in low-cost countries like India, Czech Republic, and Mexico. The deployment only in those offices would save around 6 mln \$/year. Considering that the tool can be used on different projects in various offices simultaneously and, by capturing mistakes timely, the company is saving material cost, construction work hours, and downstream discipline hours, the total savings can be much higher. Using the estimation above, it is easy to calculate the return of investment for the DL model and software development.

5.2 | Potential scale-up possibilities

The developed AI-based software can be continuously retrained to capture new typical mistakes. Thus, the tool has a scale-up possibility in the long term. McDermott process engineers identified 10 typical errors that the developed software can potentially capture. The next step is to apply the same principle to identify and capture mistakes on piping diagrams—so-called Isometrics (ISOs) (Figure 16). On McDermott's typical project, the number of ISOs issued during the EPC project is around 10 times more than the number of P&IDs. Therefore, the rough cost savings are 10 times higher and could reach 60 mln \$/year.

6 | DISCUSSION

In this research, we develop a DL model that captures typical engineering design errors and, therefore, can substantially reduce the engineering hours by lowering manual work during a quality check of P&IDs. Spotting timely various design errors by the model helps McDermott to improve the overall quality of industrial plant design and reduces the total project costs. The developed model could be potentially applied to any EPC company building complex installations like oil and gas or petrochemical plants because the design errors it captures are common within this industry.

We based our research on accumulative knowledge in engineering drawings digitalization mentioned in Section 2, which is based on symbol detection and classification by DL algorithms. In our work, we showed that the same algorithms could also be successfully adapted to recognize the design mistakes, which are, in fact, a combination of symbols on the drawings that forms a specific pattern. So far, similar solutions are neither available on the market nor found in academic literature. Therefore, it is an innovative application of existing knowledge that we would like to share with the research community and industry practitioners because the idea is domain agnostic. The process of identification of such patterns in this research case can be used as a guide for other developers in different engineering fields.

6.1 | Key findings

We also would like to highlight key findings from which other practitioners and researchers can benefit. If other practitioners decide to build a similar product, it should be noted that there are no domain datasets existing for diagrams such as P&IDs where symbols of different standards are collected.[27] Meantime, the literature review[42] and our research case emphasize that a high-quality training/test set in noisy real-world applications is crucial for the DL model performance.

We also find it essential to consider the potential scale-up possibilities. Analyzing the possible scalability helps to estimate the benefits for the company in a long term. The way it is done for this research could be an example for others who are developing similar products.

6.2 | Future work

Further research on making object detection algorithms better in dealing with small objects on busy technical drawings is required. Alternatively, splitting up the drawing into sizes suitable for the model might help to avoid significant usage of computational capacity. Thus, a large diagram can be divided into smaller rectangular regions. All the components, in this case, would be zoomed out before sending to the DL model for processing, making it easier for the model to detect patterns. After analysis by the model, results could be recomposed into the original image. This approach assumes that the resolution of the original PDF is good enough to allow zooming out. In this case, the model would have to make multiple rounds to detect the components in each rectangular sub-diagram. In this approach, it is recommended to try again the MobileNetSSD object detection and EfficientDet TensorFlow 2 models to explore more about their performance with the P&ID dataset. One-stage detector like RetinaNet, which has proven to work well with dense and small-scale objects, is also recommended to be considered in future work.[50] Additionally, the new revision of YOLO model could be considered as it is proven to be significantly smaller, faster to train and, therefore, would be more accessible for practitioners to use.[51] Future work could also include accuracy comparisons between those models.

A non-neural network approach could also be explored. We advise considering an OpenCV template matching, which finds a smaller image (template) in a larger image. It uses a sliding window method. The window slides over the source image (the image in which we want to find the template) while comparing each patch with the template. The sliding window patch that returns the best result is considered as an output. Although this method is easy to implement, it might fail to generalize properly due to minor differences in visual appearances between symbols across diagrams and requires a vast templates library to perform well.[52]

7 | THREATS TO VALIDITY

In this section, we discuss validity, that is, the potential weaknesses in the study design and the attempts to mitigate these threats.

7.1 | Internal and external validity

Although the data collected to develop the training and validation sets are from various projects executed for various international clients, they all belong to one engineering EPC contractor. The research extension in several companies would be advisable to consider in future work. However, the availability of the dataset and, therefore, a reproducibility of the results are serious threats to validity due to the confidentiality of the P&IDs. Future work should explore non-disclosure agreement possibilities with engineering companies and the adoption of confidential computing services.

7.2 | Reliability

Researcher bias could be reduced through critical reflection and by validating the lessons learned through replication.[36] To prove the reliability of this research and eliminate possible biases, it would be advisable if other researchers replicate this study to see whether a similar result could be derived.

8 | CONCLUSION

With the work hours of specialists getting more costly, the demand for automating repetitive work is increasing for EPC companies. Many organizations, therefore, are considering adopting AI to release highly skilled and expensive personnel for more complex tasks while improving efficiency and time effectiveness. In this article, we have provided a comprehensive view of AI-based solution development in order to automate the manual check of engineering complex blueprints while maintaining the quality of experience engineers' performance. It began with motivation on how to reduce the engineering hours without negatively impacting the quality. This paper discusses a particular case within McDermott where the research team used a pre-trained YOLO neural network and large training datasets with augmented patterns to detect wrong design on complex drawings. The developed model seems to be very promising in helping to reduce manual work at McDermott. Our results show that the developed model is appropriate because it can accurately find a pattern from high-level objects representing design mistakes. We also discussed the cost aspect, a potential scale-up, and key takeaways to replicate the experience of this case within the engineering industry.

We aim to share the successful experience of an EPC company's internal application development that proves that the engineering hours could be substantially reduced with the help of AI and, therefore, reinforce the EPC project's engineering phase to reduce the project's overall costs. This research has been performed on process and instrumentation diagrams, but the model is domain agnostic and can be trained on different types of schematic diagrams. The results can also motivate researchers to conduct similar application development within the engineering industry. Overall, we feel that our research can be used as a reference guide for future research and development by academics and industry professionals.

ACKNOWLEDGMENTS

This work is supported by McDermott International and Software Center (Gothenburg, Sweden). We also thank McDermott's Artificial Intelligence team members who performed the research.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

ORCID

Rimma Dzhusupova https://orcid.org/0000-0003-3411-4084
Helena Holmström Olsson https://orcid.org/0000-0002-7700-1816

REFERENCES

1. Koroteev D, Tekic Z. Artificial intelligence in oil and gas upstream: trends, challenges, and scenarios for the future. *Energy and AI*. 2021;3.
2. Arpteg A. Software engineering challenges of deep learning. In: *44th Euromicro Conference on Software Engineering and Advanced Applications*. IEEE; 2018:50-59.
3. Simon F, Christopher P, Bernd K. Systematic review on machine learning (ML) methods for manufacturing processes – identifying artificial intelligence (AI) methods for field application. *Procedia CIRP*. 2020;93:413-418. doi:10.1016/j.procir.2020.04.109
4. Balmer Roberto E, Levin Stanford L, Stephen S. Artificial intelligence applications in telecommunications and other network industries. *Telecommun Pol*. 2020;44(6).
5. Feldt R, Magazinius A. Validity threats in empirical software engineering research - an initial survey. *SEKE*. 2010;374-379.
6. Thorsten W. Machine learning in manufacturing: advantages, challenges, and applications. *Prod Manuf Res*. 2016;4(1):23-45. doi:10.1080/21693277.2016.1192517
7. Massimo B, Davide M, Mattia N, Francesco Z. Machine learning for industrial applications: a comprehensive literature review. *Expert Syst Appl*. 2021;175:114820.
8. Rimma D, Jan B, Helena HO. Challenges in developing and deploying AI in the engineering, procurement and construction industry. In: *2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC)*. IEEE; 2022.
9. Moh SN, Shifa F. Current state mapping of the supply chain in engineering procurement construction (EPC) project: a case study. *MATEC Web Conf*. 2018;195:06015.
10. EPC contractor: financing, engineering and construction. Accessed June 20, 2022. https://esfccompany.com/en/articles/engineering/epc-contractor-engineering-and-construction/
11. International standard. Functional safety of electrical/electronic/programmable electronic safety-related systems - part 1: general requirements. IEC 61508-1. 2010.
12. Hansen S. Study on the management of EPC projects. *Int J Civil Struct Environ Infrastruct Eng Res Dev (IJCSEIERD)*. 2015;5(3):11-22.
13. Baron H. *The Oil & Gas Engineering Guide*. Editions Technip; 2018:300.
14. What is an EPC contract? Here's what you need to know. 25th November 2022. Accessed November 30, 2022. https://www.blackridgeresearch.com/blog/what-is-an-epc-contract
15. Berends TC. Contracting economics of large engineering and construction projects. 2007.
16. Mani S, Haddad MA, Constantini D, Douhard W, Li Q, Poirier L. Automatic digitization of engineering diagrams using deep learning and graph search. In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. IEEE; 2020.
17. Nurminen JK, Rainio K, Syrjänen T, Paganus N, Honkoila K. Object detection in design diagrams with machine learning. In: *Progress in Computer Recognition Systems. CORES 2019*. Advances in Intelligent Systems and Computing; 2019:977-977.
18. Blanco Jose Luis, Funch Steffen, Parsons Matthew, Ribeirinho Maria João. Artificial intelligence: construction technology's next frontier. McKinsey & Company; 2018.
19. Hongfang L, Guo L, Azimi M, Huang K. Oil and gas 4.0 era: a systematic review and outlook. *Comput Ind*. 2019;111:68-90. doi:10.1016/j.compind.2019.06.007
20. Erik M, Faridaddin V, Siavash H, Thijs E, Hans V. An artificial neural network approach for cost estimation of engineering services. *Int J Constr Manag*. 2019.
21. Putra G, Triyono R. Neural network method for instrumentation and control cost estimation of the EPC companies bidding proposal. *Proc Manuf*. 2015;4:98-106. doi:10.1016/j.promfg.2015.11.019
22. Abioye SO, Oyedele LO, Akanbi L, et al. Artificial intelligence in the construction industry: a review of present status, opportunities and future challenges. *J Build Eng*. 2021;44.
23. Haun Eric. Predictive maintenance. Marine News, 2019. https://www.oedigital.com/news/472152-predictive-maintenance
24. So-Won C, Eul-Bum L, Jong-Hyun K. The engineering machine-learning automation platform (EMAP): a big-data-driven AI tool for contractors' sustainable management solutions for plant projects. *Sustainability*. 2021;13(10384):1-33.
25. Panikkar Rohit, Saleh Tamim, Szybowski Maxime, Whiteman Rob. Operationalizing machine learning in processes. 2021.

26. Rahul R, Paliwal S, Sharma M, Vig L. Automatic information extraction from piping and instrumentation diagrams. *IEEE ICPRAM 2019*. 2019.
27. Moreno-García CF, Elyan E, Jayne C. New trends on digitization of complex engineering drawings. *Neural Comput Applic*. 2019;31(6):1695-1712. doi:10.1007/s00521-018-3583-1
28. Moreno-García CF, Elyan E, Jayne C. Heuristics-based detection to improve text/graphics segmentation in complex engineering drawings. *Commun Comput Inf Sci*. 2017;87-98. doi:10.1007/978-3-319-65172-9_8
29. Ghadekar P, Joshi S, Swain D, Acharya B, Pradhan MR, Patro P. Automatic digitization of engineering diagrams using intelligent algorithms. *J Comput Sci*. 2021;17(9):833-838. doi:10.3844/jcssp.2021.833.838
30. Fagan A, West G, Mcarthur S. Human-in-the-loop approach to digitization of engineering drawings. In: *Scottish Informatics Computer Science Alliance XAI 2021 Workshop*. CEUR-WS.org; 2021:48-55.
31. Moon Y, Lee J, Mun D, Lim S. Deep learning-based method to recognize line objects and flow arrows from image-format piping and instrumentation diagrams for digitization. *Appl Sci*. 2021;11(21):10054-10054. doi:10.3390/app112110054
32. Kang SO, Lee EB, Baek HK. A digitization and conversion tool for imaged drawings to intelligent piping and instrumentation diagrams (P&ID). *Energies*. 2019;12(13):2593-2593. doi:10.3390/en12132593
33. Krizhevsky A, Sutskever I, Hinton G. ImageNet classification with deep convolutional neural networks. *Commun ACM*. 2017;60(6):84-90. doi:10.1145/3065386
34. Yun D-Y, Seo S-K, Zahid U, Lee C-J. Deep neural network for automatic image recognition of engineering diagrams. *Appl Sci*. 2020;10(11):4005. doi:10.3390/app10114005
35. Walsham G. Interpretive case studies in IS research: nature and method. *Eur J Inf Syst*. 1995;4(2):74-81. doi:10.1057/ejis.1995.9
36. Coughlan P, Coghlan D. Action research for operations management. *Int J Oper Prod Manag*. 2002;22(2):220-240. doi:10.1108/01443570210417515
37. Easterbrook S, Singer J, Storey MA, Damian D. Selecting empirical methods for software engineering research. In: Shull F, Singer J, Sjøberg DIK, eds. *Guide to Advanced Empirical Software Engineering*. Springer; 2008:285-311.
38. Soiferman Lisa Karen. Compare and contrast inductive and deductive research approaches. ERIC Number: ED542066; 2010.
39. Dodge SF, Lina K. Understanding how image quality affects deep neural networks. In: *Eighth International Conference on Quality of Multimedia Experience*. IEEE; 2016:1-6.
40. Shahinfar S, Meek P, Falzon G. How many images do I need? Understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring. *Eco Inform*. 2020;57(101085):101085. doi:10.1016/j.ecoinf.2020.101085
41. Buda M, Maki A, Maciej Mazurowski A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Netw*. 2018;106:249-259. doi:10.1016/j.neunet.2018.07.011
42. Northcutt Curtis G., Athalye Anish, Mueller Jonas. Pervasive label errors in test sets destabilize machine learning benchmarks. 2021.
43. Sanders H, Saxe J. Garbage in, garbage out: how purportedly great ML models can be screwed up by bad data. *Proceedings of Blackhat*. 2017.
44. Carneiro T, Medeiros da Nóbrega RV, Nepomuceno T, Bian GB, De Albuquerque VHC, Filho PPR. Performance analysis of Google Colaboratory as a tool for accelerating deep learning applications. *IEEE Access*. 2018;6:61677-61685. doi:10.1109/ACCESS.2018.2874767
45. Liu W, Anguelov D, Erhan D, et al. Ssd: single shot multibox detector. In: *European Conference on Computer Vision*. Springer; 2016:21-37.
46. Bochkovskiy Alexey, Wang Chien-Yao, Liao Hong-Yuan Mark. Yolov4: optimal speed and accuracy of object detection. 2020.
47. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE; 2016:779-788.
48. Srivastava S, Divekar AV, Anilkumar C, Naik I, Kulkarni V, Pattabiraman V. Comparative analysis of deep learning image detection algorithms. *J Big Data*. 2021;8(66):66. doi:10.1186/s40537-021-00434-w
49. Sculley D, Holt G, Golovin D, et al. Hidden technical debt in machine learning systems. *NIPS*. 2015;2503-2511.
50. Ahmad M, Abdullah M, Han D. Small object detection in aerial imagery using RetinaNet with anchor optimization. In: *2020 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE; 2020:1-3.
51. Benjumea A, Teeti I, Cuzzolin F, Bradley A. YOLO-Z: improving small object detection in YOLOv5 for autonomous vehicles. *ArXiv*. 2021;abs/2112.11798.
52. Mani S, Haddad M, Constantini D, Douhard W, Li Q, Poirier L. *Automatic Digitization of Engineering Diagrams using Deep Learning and Graph Search*; 2020:673-679. doi:10.1109/CVPRW50498.2020.00096

How to cite this article: Dzhusupova R, Banotra R, Bosch J, Olsson HH. Using artificial intelligence to find design errors in the engineering drawings. *J Softw Evol Proc*. 2023;e2543. doi:10.1002/smr.2543