RESEARCH PAPER

# ISA-GUIDED DIGITIZATION OF INDUSTRIAL P&ID DIAGRAMS INTO FLOW-CONSISTENT DIGITAL GRAPHS

**Mrs. R. Suganya[1], Ms. C. Vishnu Priya[2], Kamesh R[3], Nivas R[4]**
[1]Assistant Professor, Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur.
Email: suganyavsb2016@gmail.com (Corresponding Author)
[2]Assistant Professor, Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur.
Email: vishnupriyaeswari006@gmail.com
[3]UG Scholar, Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur.
Email: kameshrasu@gmail.com
[4]UG Scholar, Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur.
Email: nivaskarur04@gmail.com
***Corresponding author: Mrs. R. Suganya, Assistant Professor, Department of Artificial Intelligence and Data Science, V.S.B Engineering College, Karur**
**Email: suganyavsb2016@gmail.com**

**Received:** 30th May, 2026; **Revised:** 11th June, 2026; **Accepted:** 15th June, 2026; **Available Online:** 18th June, 2026

*ABSTRACT*

### Background

Piping and Instrumentation Diagrams (P&IDs) show how industrial processes are laid out, including piping systems and control instruments. Many old P&IDs are just scanned images or static PDFs, which makes it hard to integrate them into current digital workflows like process simulation, predictive maintenance, and digital twin systems. Manually converting them into digital formats takes a lot of time and can lead to mistakes. Existing automated solutions often struggle with different drawing styles, high-resolution files, and complex pipe junctions.

### Objective

This paper presents a mixed computer vision approach to change different P&ID drawings into structured, ISA-5.1-compliant JSON graphs.

### Materials and Methods

The method starts with preprocessing guided by standards. This includes binarization, skeletonization, and morphological operations to standardize line styles and fix scanning issues. To improve small object detection in large diagrams, Slicing Aided Hyper Inference (SAHI) divides high-resolution images into overlapping sections. This enhances component detection based on the YOLO model. A fine-tuned YOLO model, trained on expanded industrial datasets, identifies valves, pumps, and instruments across different vendors. Text tags are extracted and linked to the detected components to clear up any layout confusion. Pipe structures are found using edge detection and Hough transforms. Connectivity is established through directional continuity to form junction-aware topology.

### Results

The final result is a structured graph representation that includes nodes and directed edges, making it suitable for industrial analytics applications.

**Index Terms:** Piping and Instrumentation Diagrams (P&ID), ISA-guided digitization, component detection, flow-consistent graph construction, junction-aware connectivity, SAHI-based inference, vision-based label extraction, digital graph generation, digital twin.
**How to cite this article:** Suganya R, Vishnu Priya C, Kamesh R, Nivas R. ISA-Guided Digitization of Industrial P&ID Diagrams into Flow-Consistent Digital Graphs. Int J Drug Deliv Technol. 2026;16(60s):1878-1885. DOI: 10.25258/ijddt.16.60s.167
**Source of support:** Nil.
**Conflict of interest:** None

## INTRODUCTION

P&IDs, as they are commonly referred to in the engineering world, are more than just pictures on a wall in a control room. P&IDs are detailed representations of an industrial process. A P&ID can show how pumps move fluids through a series of pipes, valves controlling the fluid flow, storage containers holding substances, and monitoring devices feeding information into a system. If you examine a P&ID closely, it is not just a picture but a language. In refineries, chemical plants, and even power plants, P&IDs are considered legal documents. They direct the construction process. They show where everything is connected as a technician studies them during a maintenance operation. Inspectors use them during a safety audit to ensure the correct placement of safety valves. They could be said to be the memory of the plant.

Despite the advances made in modern computer design software, many of these P&IDs still exist as old documents stored as images.

They are easy for an engineer to read, but a computer only understands pixels. This is an increasingly important difference.

### A. The Need for Structured Digitization

Modern industrial systems heavily rely on computing technologies. Simulation systems, predictive maintenance systems, and digital twins all need structured representations. If we consider this from a computing perspective, a plant is a graph. We can consider equipment as nodes, while pipes can be considered as edges. We can also consider the direction of the flow as an orientation for the edges.

However, the challenge is that this conversion from existing diagrams is still a very manual process. We can consider an engineer looking at a scanned diagram, identifying symbols one by one, looking at the pipelines visually, and entering relationships into a system. This is a process that demands both time and knowledge. If we miss a branch or misinterpret a junction, we end up with a wrong representation in the system.

Although this might be manageable for a small pilot plant, it is not realistic for a refinery plant with potentially hundreds of P&IDs. Of course, there is a strong interest in automating the reconstruction of topology from image-based diagrams. The reason for this is obvious.

### B. Technical Difficulties in Automating P&ID Digitization

At first glance, this problem seems quite simple: find symbols, find lines, connect them. Well, it is not as simple as it seems.

In the first place, P&ID diagrams in the real world do not follow a single pattern. Even if a certain organization adheres to a set of rules, such as ISA-5.1, symbols still do not match perfectly. A symbol for a control valve drawn by one engineer may not match exactly a symbol for the same component drawn by another engineer.

Another factor is the use of scanned documents. Lines may be dark in one place and light in another. Some lines may be broken. In addition, background noise is also a problem when documents are copied multiple times. Compression of images also affects thin lines of pipelines, transforming them into irregular shapes.

One more factor is resolution. P&ID diagrams in the real world can be quite large. Dozens of elements can be packed into a single diagram. A small symbol for a pressure transmitter may be only a small fraction of the whole document.

Lastly, the reconstruction of connectivity is probably the most difficult task. Pipes may cross over each other, split into two or more directions, and merge into a single pipe. Misinterpreting a junction can change the topology of the whole process. This is not a minor error; it is a serious mistake in a safety-critical system.

### C. Limitations of Existing Methods

Typically, research in this area has followed two paths. One line of research uses a modular design. That is, one model is used to detect symbols, another to detect text, and a third to detect lines. The results are then combined to produce a final graph. The benefit of this approach is that it is easy to understand. It is easy to debug and improve each module individually. However, the drawback is that it is equally easy to see the limitations. If a symbol is not detected early on, there will be nothing to attach the connectivity results to. If a line is in several pieces, the final graph will reflect this**.**

The second line of research uses end-to-end learning models. Typically, this means using a transformer architecture. That is, the model attempts to simultaneously detect objects and their inter-relationships. The benefit of this approach is that it can learn structural patterns directly from the input data. However, it requires a lot of data and a lot of computational power. High-resolution images need to be subdivided into smaller sections. However, once this is done, problems arise. If a symbol is close to the edge of a section, it will be detected twice. Additional logic is needed to handle this. Both lines of research have their benefits. However, it seems that there is still room for a solution that combines the benefits of the learning-based approach and the modular approach in a way that remains feasible in the industrial world.

### D. A Hybrid, Topology-Oriented Approach

This approach is based on finding the middle ground by using the power of deep learning for detection, along with rule-based reasoning for the topology.

We first apply ISA-5.1 guided preprocessing, i.e., standardizing the look of the lines, as well as removing as much scan noise as possible. For the problem of small objects inside large diagrams, Slicing Aided Hyper Inference divides the diagram into overlapping sections while preserving resolution. We apply the YOLO model, extended for the expanded industrial symbol data set, for detection inside the sections created by the Slicing Aided Hyper Inference.

However, the detection of the pipe follows a different set of rules, i.e., we apply skeletonization as well as Hough line detection, along with connectivity by checking directional continuity. This is analogous to saying that the detection of the pipe is not merely detection, but rather tracing the object. Similarly, the detection of the junction is not based on guessing, but rather by checking the continuity of the object.

The resulting output is exported as a structured JSON graph, i.e., the nodes contain the spatial coordinates as well as the attributes of the component, while the edges contain the direction of the connections. This is analogous to saying that the diagram is no longer an image, but rather data that can be fed into the simulation environment, thus providing the digital twin.

### E. Contributions

The contributions of this work as follows:

1. A preprocessing strategy designed to improve robustness against noisy and degraded scanned P&IDs.
2. A high-resolution detection framework combining slicing techniques with YOLO to better capture small-scale symbols.
3. A junction-aware topology construction method based on directional continuity rules rather than purely learned associations.
4. A structured JSON graph output that supports direct use in computational and simulation systems.

## RELATED WORK

There has been a steady increase of interest in the automation of the digitization of Piping and Instrument Diagrams, and rightly so, as the structured data representation of a P&ID can be used to drive simulators, asset management tools, and digital twins without the need to have someone draw arrows on a computer screen. This has captured the interest of computer vision, document analysis, and even graph theory experts. Over the course of the last ten years or so, there have been suggestions that go from very rule-based approaches to fully end-to-end transformer approaches that attempt to go from the pixels to the graph directly.

However, despite all the interest and effort, the problem still has not gone away, and connectivity issues remain, as do the issues of vendor-specific drawing oddities.

### A. Modular and Pipeline-Based Approaches

Automatic Digitization of Piping and Instrumentation Diagrams [1] by Shubham Paliwal, et al., was the first attempt at a structured approach to the problem of P&ID digitization. Their approach was a very linear, almost ‘textbook-style’ approach to the problem. They used convolutional neural networks to identify symbols, kernel-based image processing to identify the piping, optical character recognition to identify the textual components, and a set of rules to map symbols to the lines.

In order to train the model, the authors have used a synthetic dataset of P&ID diagrams with symbols labelled. This was a good approach, as industrial diagrams aren’t typically publicly available.

The approach of the authors was to convert the diagrams into a more machine-understandable form, which was a big step in itself.

However, the approach of the authors to map symbols to lines was a very proximity-based approach. If a valve symbol was close enough to a line, the model would assume a connection.

This approach would work well if the diagrams aren’t too complex, but if there are a lot of parallel pipes, the proximity approach would be less effective. Consider a scenario in which there’s a cluster of pipelines in a refinery’s distillation unit, and a slight error would change the entire topology of the system.

A similar approach by S. Mani, K. Goyal, and R. Krishnamurthy, ‘Automatic Digitization of Engineering Diagrams Using Deep Learning and Graph Search,[2]’ was a step in the right direction, as the authors used a deep learning approach to identify the symbols, as well as the lines, and a graph search approach to determine the connections.

However, the approach of the authors to determine the connections was still based on proximity to the line, as well as a set of constraints. While this approach was better than the earlier approach of using a set of rules to determine the connections, the proximity to the line was still a weakness of the approach, as a slight variation in the image would result in a different set of connections.

### B. Template-Driven and Rule-Oriented Systems

Another branch relies more on pre-defined symbol libraries. In their paper titled "A Digitization and Conversion Tool for Imaged Drawings to Intelligent P&IDs," S. O. Kang, E. B. Lee, and H. K. [3] Baek propose a system that employs a combination of sliding window analysis, template matching, and OCR-based symbol extraction. In their system, symbols are identified by matching their images with pre-defined templates. Then, structured diagrams are constructed through rule-based logic.

In a controlled environment such as company-internal drawings with consistent symbol libraries, such a method performs reasonably well. When all symbols of a certain type, such as all control valve symbols, have similar characteristics and follow similar patterns, it can actually feel comforting to use such a method. However, in real-world scenarios, such a method will not perform optimally because contractors may modify symbols slightly, and drawings may have been scanned and may have skewed or faded images of symbols. Moreover, no matter how comprehensive a template library may be, it will never be able to cover all possible variations of symbols.

In their paper titled "Feature Recognition from Piping and Instrumentation Diagrams in Image Format Using a Deep Learning Network,[4]" Yu and his team propose a method of improving symbol recognition accuracy through the use of convolutional neural networks. While it does improve symbol recognition accuracy by moving away from pre-defined templates and features, it still relies on heuristic logic and traditional OCR-based symbol recognition and connectivity analysis. When symbols and pipes are closely grouped together, such as in compressor stations and lines feeding reactors, heuristic logic may fail. While it may have high accuracy in recognizing symbols individually, it may still not have correct topology.

### C. Deep Learning and End-to-End Architectures

Recently, there has been a shift in focus to end-to-end solutions. In "Transforming Engineering Diagrams: A Novel Method for P&ID Digitization," J. Stürmer, M. Graumann, and T. Koch [5] introduce a transformer architecture that views P&ID digitization as an image-to-graph problem. The Relation transformer is used to generate bounding boxes and relational edges, and the PID2Graph dataset provides a graph annotation that can be used to evaluate the model’s performance in reconstructing the topology.

This reduces the need for post-processing, as the edge prediction and object detection occur together. The accuracy in edge prediction has been shown to exceed that of modular solutions in the experiment. However, there is a drawback to using such a model. It requires a lot of data to train and a lot of computational power. Typically, industrial images are in high resolution and are divided into smaller sections to be processed. However, this can cause issues as the edges can cause problems in the final model. Objects that are close to the edge may not be recognized as complete objects and may be counted twice, requiring additional processing to merge them into a single object. Thus, the advantages of using an end-to-end model are offset by the problems that can occur.

In "End-to-End Digitization of Image Format Piping and Instrumentation Diagrams [6]," B. C. Kim et al. introduce a model that combines the processes of object detection, topology reconstruction, and digital export into a single system. The topology is reconstructed using the snapping of endpoints and a series of rules. The results can be exported into a 3D system.

This has the advantage that it can be used in a 3D system, which can be beneficial in the export process. However, there are problems that can occur in the process. If there are several pipes that cross each other, the endpoints will be snapped incorrectly. Similarly, if there is a problem in the process, it can cause the entire process to be inaccurate, giving a false representation of the process flow.

### D. Recurring Limitations

When examining the variety of approaches, there are recurring limitations and difficulties to work around.

There is a significant degree of reliance on the distance between things in connectivity methods. Template-based methods heavily depend on the quality of the symbol libraries provided.

Deep learning methods improve the object detection process but often rely on geometric thresholds to connect the pieces. Even the transformer-based methods reduce the amount of rules and thresholds but require large datasets and significant computing power, which not all industries can afford.

There is also the issue of dealing with densely filled diagrams at high resolutions. The ability to identify small objects and maintain the context is critical. The meaning of the junctions also has to be determined and not simply estimated.

It can be said that none of the methods have effectively addressed all the limitations and issues. However, this also speaks to the benefits of hybrid approaches and the ability to effectively combine the benefits of small object detection, the ability to process and prepare data in accordance with the industry requirements, and the use of deterministic topology methods to address the junctions. Perhaps the way forward is not necessarily the use of rules and machine learning but the ability to effectively combine the two.

## II. PROPOSED SOLUTION

Now, I'll try to walk you through just how this system is meant to work in practice, not just as a series of steps but as a living process. The idea behind it is deceptively simple but surprisingly difficult: take a P&ID diagram as a static image—in whatever form it happens to be received, be it a scanned image from a panel, a crooked PDF, a CAD export, etc.—and transform it into something a machine can reason about. Not just recognize symbols for what they are, but understand their structure. Not just see lines connecting symbols, but follow the flow and understand it.

This system is designed with this end in mind. It's a combination of standards-aware preprocessing, high-resolution learning-based detection, context-aware label detection, classic pipe detection, and a more thoughtful approach to topology reasoning. It's a delicate balance between allowing machine learning to do its thing while also ensuring deterministic rules for structural understanding. The idea is to get robustness without sacrificing interpretability. In an industrial context, errors aren't allowed to be mysterious.

### A. The System at a Glance

The process can be thought of as a step-by-step process. Each step has something unique to do. So, we start with the rough drawings and gradually refine them to a clean and well-structured graph, the process begins with the preprocessing stage and the ISA. Real P&IDs aren't very clean. Some diagrams have thick and bold pipe lines. Others have faint and thin pipe lines. Similarly, the scanned diagrams also vary. Some diagrams have very clear scans, while others have scans that have deteriorated over time and have the faint signs of having gone through the photocopier process multiple times. So, before the actual process begins and the AI model starts to act, these diagrams are made clean and normalized.

Next, we have the high-resolution inference process using the SAHI – Slicing Aided Hyper Inference process. Instead of the entire diagram being shrunk to fit the dimensions required by the neural network model, the diagram is divided into sections. Each section has the same level of detail and precision. For example, the small 'flow indicators' would not get blurred during this process.

Next, the identified sections are passed through the YOLO model to identify the symbols. Each identified object is then scanned to identify the semantics using the localized vision-based label extraction process.

At the same time, the pipe diagrams are extracted using the image processing techniques. Finally, the entire process culminates in the graph building stage. So, the diagram isn't simply an image anymore; it has turned into data.

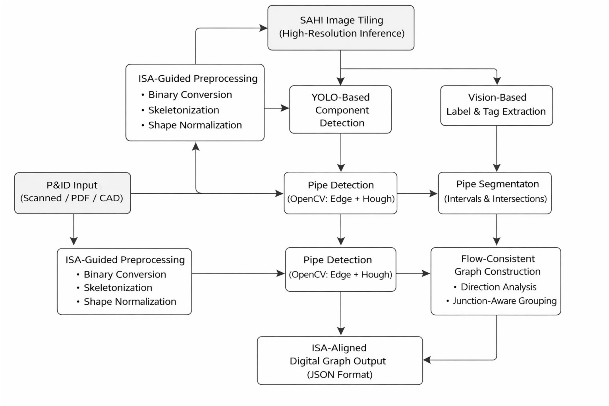


Fig 1: System Flow

### B. ISA-Guided Image Preprocessing

Industrial P&IDs do not have clean and regular shapes. Line thickness changes, contrast changes from page to page, and symbols become blurry but to varying degrees. Applying the model directly to these images would degrade the results – not because the model is bad, but because the data fed to the model is inconsistent.

To mitigate this problem, the preprocessing follows a simple and structured methodology based on ISA-5.1. The steps are simple, but they are executed in a specific order.

Firstly, the image is converted to grayscale. The data in the image does not contain useful colour information and could introduce noise. Then adaptive thresholding is applied to segment the structural lines and the background. Morphological operations such as erosion and dilation are used to remove small details.

Skeletonization is also critical. All the lines are made to have a thickness of one pixel. This might seem extreme, but the reasoning behind this is to address the problem that the thickness of the lines in the P&ID could vary. When the lines are thinned to the same thickness, the difference between the thickness of the lines becomes negligible.

At the end of the preprocessing steps, these mixed drawings start to resemble a clean and canonical structure. The model is not made simpler; the data is made much simpler for the model to process.

### C. High-Resolution Inference with SAHI

The size of the industrial P&IDs can be in the thousands of pixels across the image. It is not feasible to input the entire image into the network because the image is too big to be input into the network, which has limited input size. If the image is shrunk to input into the network, the small features in the image will not be visible. A small pressure transmitter will become a blur, and the arrow showing the flow will not be sharp.

SAHI addresses this issue by dividing the image into tiles and processing them individually at their native resolution. This way, the small features in the image are not distorted and remain sharp. After the inference is complete, the results from all the tiles are combined into global coordinates using confidence-based non-maximum suppression. This results in the creation of a single set of bounding boxes from the image at its native scale.

### D. Learning-Based Component Detection

Our learning-based component detection model is based on the YOLO algorithm, which has been trained on symbol datasets that conform to the ISA standard. To enhance the model's robustness, we incorporate data augmentation techniques such as rotation, scaling, brightness, and noise to mimic the real-world environment. It is common knowledge that industrial drawings are seldom perfect.

Fig 2: Illustrates a representative detection result obtained using SAHI-based inference, showing reliable identification of components across dense regions of the P&ID.

Our model detects items such as valves, pumps, tanks, instrumentation symbols, flow indicators, and control items. Unlike template-based methods that compare the image patches to a symbol library, the YOLO algorithm learns the unique geometric features directly from the data. This feature is particularly useful when vendors introduce minor stylistic variations in the drawing.

Our model produces class label information, confidence information, and bounding box coordinates for each detected item, which will be used as nodes in the reconstructed graph.

### E. Contextual Label Extraction

After the components have been identified, the next step is to anchor the correct text labels. When the whole page is processed using OCR, the results usually backfire. The tags get mixed up, the text blocks overwhelm the neighbouring symbols, and the relationships get confusing.

So instead, the process crops the bounding box and performs the label extraction. It also performs spatial reasoning to filter the results since the object's location isn't the only determining factor. It also factors in the alignment and context.

For instance, if there is a tag such as "FV-101" that is slightly above the valve symbol, the relationship would differ if the same tag is placed between two parallel pipelines. The contextual filtering improves the semantic results.

### F. Pipe Detection and Segmentation

For pipe extraction, we treat this as a separate process to keep the model simple. First, edges are used to identify possible pipe locations. Then, skeleton thinning is applied to the edges to reduce the edges to centre lines. This produces clean and linear shapes. A probabilistic Hough Transform is used to find straight edges.

Next, the edges are combined to form continuous pipelines. To simplify topology reasoning, the pipelines are broken down into atomic pieces at intersections and at fixed intervals. This breaks down the pipelines to reveal the topology of the connections.

Dashed and solid lines can also be distinguished by examining the continuity and intervals. This is important to identify whether the lines represent signal flows instead of physical pipes.

### G. Flow-Consistent, Junction-Aware Graph Construction

We establish connectivity by means of deterministic structural reasoning, during which we assign directional information based on the orientation of the pipe segment's geometry.

We don't connect elements by checking for proximity; rather, we check for the consistency of direction. Elements that share directional consistency form an uninterrupted flow path. By examining the intersection cluster, we can determine the location of the junction nodes. The endpoint of the pipe is connected to the boundary of the component when the two share consistency.

This technique reduces the incidence of incorrect connections, especially when the layout is dense. T-junctions and branch structures are also taken into consideration. There is active filtering of incorrect connections, as this can impact the topology of the process.

The identified components become the nodes of the graph, while the correct connections between the elements become the edges.

### H. Structured ISA-Aligned Graph Output

The final output is a well-structured JSON graph that is in compliance with the ISA format. Each node has its own unique identifier, the class of the component, the bounding box coordinates, the tag information if extracted, and spatial information relevant to the graph. The edges have information on where they start and end, the direction of the edge, and whether it is a solid or dashed line.

This output is quite natural and flows well with the digital twin, simulators, process simulators, asset monitoring, and analytics tools. More importantly, it is well-aligned with the ability to reason through the process topology without the need to manually intervene. The big idea here is that the system is not locked into either pure learning or pure rules-based logic. It is able to recognize what is present in the scene through the use of data-driven recognition and then reason through how these pieces connect in a structured fashion. This is quite important in the industrial world, where misinterpretation of a single junction in the process can have disastrous consequences further down the process.

To illustrate the output of the proposed framework, a sample JSON representation of the generated graph is shown below

```
{
  "nodes": [
    {
      "id": "77ebee18-0fa6-497b-aa45-fe415a88ca06",
      "label": "Ball Valve",
      "type": "22",
      "bbox": [247.5, 286.5, 266.5, 297.5]
    },
    {
      "id": "b3811cbe-58ea-416d-9987-50d7a604d039",
      "label": "Control Valve",
      "type": "28",
      "bbox": [405, 76.5, 423, 87.5]
    }
  ],
  "edges": [
    {
      "from": "60a87cc9-ead9-46f4-ab73-a47f1f2c5ac4",
      "to": "b3811cbe-58ea-416d-9987-50d7a604d039",
      "length": 420
    }
  ]
}
```

Fig 3: Example JSON graph output generated by the proposed framework.

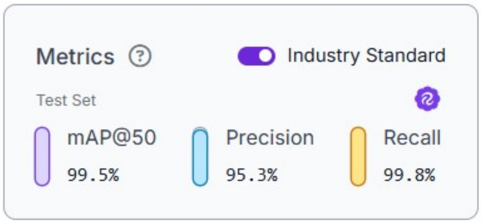
## RESULTS AND DISCUSSION

### A. Detection Performance Evaluation

The component detection model, which is an extension of YOLO, has been trained and tested through the Roboflow platform. Metrics such as Precision, Recall, and Average Precision at 0.5 IoU (mAP@50) have been used to evaluate the model on the test set.

The trained model achieved:

1. **mAP@50: 99.5%**
2. **Precision: 95.3%**
3. **Recall: 99.8%**



Metrics

Industry Standard

Test Set

mAP@50 99.5%

Precision 95.3%

Recall 99.8%

Fig. 4. Detection performance metrics on the test dataset (mAP@50, Precision, Recall).

The recall is extremely high, indicating that the model is able to recognize almost all relevant industrial components in the test set. With the precision being 95.3%, the false positives are minimized, although some errors may occur for the symbol types that look similar to each other.

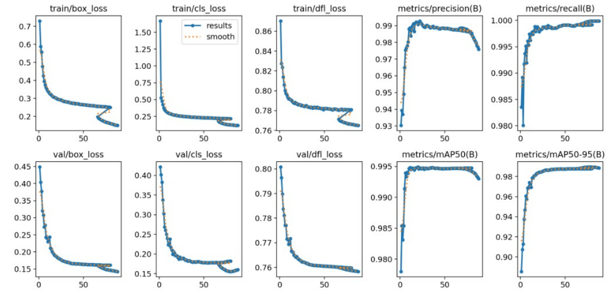


Fig. 5. Training and validation performance curves showing box loss, classification loss, DFL loss, precision, recall, mAP@50, and mAP@50–95 across training epochs.

### B. Training Stability and Convergence Analysis

The training curves also indicate the detector's behaviour. The model seems to learn quickly and then stabilizes smoothly. At the beginning of the training process, the performance of both mAP@50 and mAP@50–95 increases rapidly. This indicates the model is quickly picking up the visual cues for the industrial symbols. Then the performance stabilizes around 99.5%, and there are no significant bumps and dips.

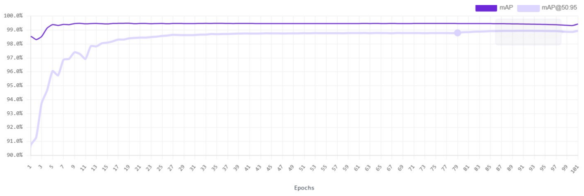


Fig. 6. mAP@50 and mAP@50–95 convergence curves across training epochs.

The slightly worse performance in mAP@50–95 with tighter IoU thresholds also indicates the model's ability to keep the predicted bounding boxes well aligned with the ground-truth bounding boxes.

It's also worth noting there is no significant drop in the validation set's performance during the training process. This again indicates the model is well balanced in object localization and object classification.

The convergence behaviour also indicates the model is well optimized and the generalization performance is reliable within the validation set.

### C. Impact of High-Resolution Inference Using SAHI

The industrial P&ID diagrams are large diagrams, with many symbols very close to each other. If we try to reduce the whole image to the expected input size of a neural network, the details of the image get lost

The difficult part of the image consists of the small instruments around the intersection of the pipes, which can be easily missed.

With the use of SAHI, the large diagrams are divided into tiles, which are then processed separately before the final image is obtained by overlapping the tiles. This way, the details of the image, such as the small symbols, remain clear instead of becoming a blur.

This improves the recall of the image, especially the small instrumentation components, which need to be correctly identified to form a correct image.

### D. Graph Reconstruction and Topology Validation

Identifying the symbols is not the full task, however. The P&ID diagram is not complete until the connections between the different parts are understood. Once the different parts of the diagram were identified, the extracted pipe segments were used to recreate the process as a graph.

Unlike the case when all the pipes are treated as continuous lines, the program divides the pipelines into smaller sections. This makes the identification of the different junctions easier. The decision on the connections between the different parts is not solely based on the proximity of the different parts. The program also considers the directional continuity between the different parts of the pipe and the different parts of the components. This is important when the diagram is densely drawn, such that the different parts of the pipe may be crossing or may be parallel to the different parts of the component.

As the results of the topology validation, the different components of the diagram were easily identified as the different nodes, while the connections between the different parts of the diagram, as validated, were easily identified as the edges of the different graphs. The identification of the different junctions was also easy, while the direction was maintained for most of the cases, unlike the results that could be obtained by using the basic proximity methods, whereby the resulting graphs were more coherent.

There are, however, some difficulties that may arise, especially when the quality of the scanned diagram is not good, as well as when the diagram is not drawn clearly. The results, however, show that the topology reconstruction method is stable and dependable.

## III. CONCLUSION

This paper proposes a hybrid approach for converting the given P&ID image into a structured graph, as per the ISA standard. Instead of focusing on a single technique, we're going for the hybrid approach, allowing both methods to shine when they're applicable.

The proposed approach involves standards-aware preprocessing, high-resolution tiled inference using the SAHI model, component detection using YOLO, contextual label recognition, and topology reconstruction while considering the direction of the flow and the functionality of the junctions. After that, the graph is constructed based on the topology that has been discovered. The results look promising, and many thoughts come to the mind while seeing the results.

The detector has achieved good precision, recall, and mAP on the test set, while the learning curves increase smoothly without any sign of overfitting during the later stages of the experiment. The usage of high-resolution images has also helped us achieve the desired results, as the details of the small instrumentation symbols are retained while working with them, as they tend to blur when downsized for the entire image detection task.

The construction of the graph based on the topology has also helped us achieve good connectivity, especially when the image has many intersections or parallel pipes. The output is a structured JSON graph that represents the process. It is not just the nerdy representation of the process, as it is the operational data that could be used for many different purposes, such as simulations, digital twins, and many different types of graph analysis tools.

This is not the perfect result, as the quality of the scanned image is an issue, along with the curved pipes, for which more tests need to be performed with images from different sectors of the industry for more confidence in using this for the digitization of the P&ID image. The experiments also suggest that the right way forward for the task of P&ID image digitization is the hybrid approach that uses both machine learning detection and structured reasoning.

## VI. REFERENCE

[1] S. Paliwal, A. Jain, M. Sharma, and L. Vig, “Digitize-PID: Automatic Digitization of Piping and Instrumentation Diagrams,” in *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, 2021, pp. 168–180.

[2] S. Mani, M. A. Haddad, D. Constantini, W. Douhard, Q. Li, and L. Poirier, “Automatic Digitization of Engineering Diagrams Using Deep Learning and Graph Search,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2020, pp. 673–679.

[3] S. O. Kang, E. B. Lee, and H. K. Baek, “A Digitization and Conversion Tool for Imaged Drawings to Intelligent P&IDs,” *Energies*, vol. 12, no. 13, p. 2593, 2019.

[4] J.-M. Cha, K. Taewoo, T. Lee, and D. Mun, “Feature Recognition from Piping and Instrumentation Diagrams in Image Format Using a Deep Learning Network,” *Energies*, vol. 12, no. 23, p. 4425, 2019.

[5] J. M. Stürmer, M. Graumann, and T. Koch, “From Engineering Diagrams to Graphs: Digitizing P&IDs with Transformers,” *arXiv preprint arXiv:2411.13929*, 2024.

[6] B. C. Kim, H. Kim, Y. Moon, G. Lee, and D. Mun, “End-to-End Digitization of Image Format Piping and Instrumentation Diagrams at an Industrially Applicable Level,” *Journal of Computational Design and Engineering*, vol. 9, no. 4, pp. 1298–1315, 2022.

[7] L. Jamieson, C. Francisco Moreno-Garc´ıa, and E. Elyan, “A review of deep learning methods for digitisation of complex documents and engineering diagrams,” Artificial Intelligence Review, vol. 57, no. 6, 2024.

[8] J. K. Nurminen, K. Rainio, J.-P. Numminen, T. Syrjanen, ¨ N. Paganus, and K. Honkoila, “Object detection in design diagrams with machine learning,” in Progress in Computer Recognition Systems, ser. Advances in Intelligent Systems and Computing, R. Burduk, M. Kurzynski, and M. Wozniak, Eds. Cham: Springer International Publishing, 2020, vol. 977, pp. 27–36.

[9] C. F. Moreno-Garcia, P. Johnston, and B. Garkuwa, “Pixel-based layer segmentation of complex engineering drawings using convolutional neural networks,” in 2020 International Joint

[10] S. Paliwal, M. Sharma, and L. Vig, “Ossr-pid: One-shot symbol recognition in p&id sheets using path sampling and gcn,” 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8, 2021.