Journal of Computer Science

Original Research Paper

Automatic Digitization of Engineering Diagrams using Intelligent Algorithms

[1]Premanand Ghadekar, [2]Shaunak Joshi, [3]Debabrata Swain, [4]Biswaranjan Acharya,
[5]Manas Ranjan Pradhan and [6]Pramoda Patro

[1]Department of Information Technology, Vishwakarma Institute of Technology, Pune, India
[2]Department of Information Technology, Vishwakarma Institute of Technology, Pune, India
[3]Department of Computer Science and Engineering, Pandit Deendayal Energy University, Gandhinagar, India
[4]School of Computer Engineering, KIIT University, Bhubaneswar, India
[5]School of Information Technology, Skyline University College, Sharjah, UAE
[6]Department of Engineering Science, Amrutvahini college of engineering, Sangamner, India

Article history
Received: 16-05-2021
Revised: 02-07-2021
Accepted: 06-07-2021

Corresponding Author:
Dr. Debabrata Swain
Department of Computer Science and Engineering,
Pandit Deendayal Energy University, Gandhinagar, India
Email: debabrata.swain7@yahoo.com

Abstract: At present usage of computational intelligence became the ultimate need of the heavy engineering industries. Digitization can be achieved in these sectors by scanning the hard copy images. When older documents are digitized are not of very high fidelity and therefore the accuracy, reliability of the estimates of components such as equipment and materials after digitization are remarkably low since (Piping and Instrumentation Diagrams) P&IDs come in various shapes and sizes, with varying levels of quality along with myriad smaller challenges such as low resolution of images, high intra project diagram variation along with no standardization in the engineering sector for diagram representation to name a few, digitizing P&IDs remains a challenging problem. In this study an end to end pipeline is proposed for automatically digitizing engineering diagrams which would involve automatic recognition, classification and extraction of diagram components from images and scans of engineering drawings such as P&IDs and automatically generating digitized drawings automatically from this obtained data. This would be done using image processing algorithms such as template matching, canny edge detection and the sliding window method. Then the lines would be obtained from the P&ID using canny edge detection and sliding window approach, the text would be recognized using an aspect ratio calculation. Finally, all the extracted components of the P&ID are associated with the closest texts present and the components mapped to each other. By the way of using such pipelines as proposed the diagrams are consistently of high quality, other smaller problems such as mis-spelling and valuable time churn are solved or minimized to large extent and paving the way for application of big data technologies such as machine learning analytics on these diagrams resulting in further efficiencies in operational processes.

Keywords: P&ID Sheets, Computer Vision, Industrial Automation, Image Processing, Tree Search

Introduction

P&ID are standardized representations for depiction of equipment and process flow involved in a physical process. Many complex engineering workflows depicting the schematics of a process flow diagram through its components such as inlets, pipeline paths, symbols which represent instruments and other miscellaneous equipment's. In many engineering sectors these data rich files are often stored in the physical or scanned file format and are often archived for further use. However, there is no intelligent pipeline for these massive stores of data in order to extract and analyze this data. Any operation on these large number of files requires massive amounts of human labor and time commitment re-orienting with these files which often results in delays, incorrect analyses and cost overruns. It would be a massive boon if all this data stored away could be digitized and used to gain valuable insights

Science Publications

into the inner connections of the plant components to each other and their behavior. This would result in a large jump in engineering efficiency, cost savings and reduced use of valuable engineering manpower.

Related Work

Since the 1980s, computers were mostly used in making engineering artefacts. Computer researchers invented methodologies which transition an engineering drawing to digital forms. Brown *et al*. (1988) and Joseph (1989) brought forth Optical Character Recognition (OCR) techniques which used Boolean logic-based symbol and numerical character recognition methods and line conversion methods to create the CAD system equivalents. An approach to distinguish the text and graphics used in an image was designed by Lu (1998) that differentiates by erasing non-text areas in an image. Region-based approach which used vectorization was proposed by Chiang *et al*. (1998) for recognizing pixel ensembles within line segments. Nagasamy and Langrana (1990) applied the vectorization method to create CAD, CAM applications information of any diagram from scanned images. Kacem *et al*. (2001) implemented fuzzy logic method to extract printed mathematical formulas, algorithmically. A method to separate symbols from equations by connection lines based on generic properties of connection lines and symbols was developed by Yu *et al*. (1994). Adam *et al*. (2000) applied a technique to classify the patterns of a technical document, using Fourier–Mellin transform. General CAD conversion problems were discussed but no global application was developed. A network model was designed by Ah-Soon (1997), that identified symbols from a scanned drawing inspired by Messmer and Bunk (1995; 1997) algorithm. (Lu *et al*. (2007) used analysis of various drawings to auto reconstruct and recognize drawings. Wenyin *et al*. (2007) developed a cooperating method for graphical recognition in engineering scans while Guo *et al*. (2012) proposed example-driven symbol recognition. Wei *et al*. (2017) proposed a unique method given scene images to detect text based on segmentation. In the recent decade, intelligent algorithms which include neural networks (Gellaboina and Venkoparao, 2009), Machine intelligence techniques (Elyan *et al*., 2018) such as deep learning have been used for this.

Proposed Methodology

This study aims to automate the process of semantically understanding and digitizing engineering diagrams to achieve a faster workflow and so save costly man hours of work as shown in Fig. 1.

Text Extraction and Detection

For a scanned P&ID, we extract text elements through the Document Object Models (DOM) (https://www.w3.org/TR/2000/WD-DOMhttps://www.w3.org/TR/2000/WD-DOM-Level-1-20000929/DOM.pdfLevel-1-20000929/DOM.pdf) by traversing through the tree structure and using a regular expression to match the texts. These texts are then checked with other boxes for intersection and intersecting text boxes are merged, however in cases where a P&ID is scanned, the vectors are lost and a plain image is presented. Here, in order to find the areas where text exists, the aspect ratio is calculated and using this data the text is recognized and extracted from the respective area. OCR is a viable tool used here to recognize text. Since the P&ID has a mixture of lines, symbols and texts, due to this text recognition accuracy falls. Thus, there is a requirement for a methodology of capturing regions which hold text by using the aspect ratio of characters in the P&ID and based on this recognizing the text in the regions. To find this area first we mask the lines and instrument markings. Recognized parts are removed when they exceed the aspect ratio If it is within the preset aspect ratio then it is kept. Then, if the recognized part is determined to be a text area, a contour bounding box of the entire text area is created by leaving the text area in such a manner that the entire text area is extracted. Once this area is determined, the OCR is applied. Since, the rate of recognition is not 100%, text training is done. If the rate of recognition is lower than a threshold then the characters from the image are mapped in each image. Finally generate the training data, store in a database and apply text recognition.

Finding Candidates Lines

Each of the paths in the SVG file are linked together and checked against a line length threshold only after which, are considered for further mathematical computation.

Pipeline Matching

Distance between each text line and text object is calculated and the nearest ones are linked together subject to (1) Regex (2) line-text pair orientation and (3) distance thresholding. Corner cases such as arrow lines pointing a specific text to a pipeline and two separate lines which overlap are also handled.

Symbol Detection

Equipment Symbol detection and recognition on image P&IDs was done using template matching which is a commonly used image conversion technique. Pre-processing an image pyramid is built using symbol templates while being binarized and color inverted. This is to improve matching scores as spurious calculations which are done due to different color levels are prevented. Although in some of the P&ID the orientation of the symbols is uniform and fixed, it is often the case that the objects that are to be detected appear with a certain angle of rotation.

This is solved by computing not just one template image pyramid, but a set of pyramids - one for each possible rotation of the template coupled with the different sizes which resulted in 4 image pyramids.

Pre-Processing

An image pyramid is built using symbol templates. These templates are first converted in an image pyramid,

Wherein each image is down sampled and up sampled at different scales to create larger and smaller images of the same base image as shown in Fig. 2. Similarly, here the images to be matched are which are called templates are converted into a series of images sequentially down sampled and up sampled by a margin of 2. Each template is scaled in a range of 0.1 and 2, this creating different sizes of the same template which allows of scale invariant matching.

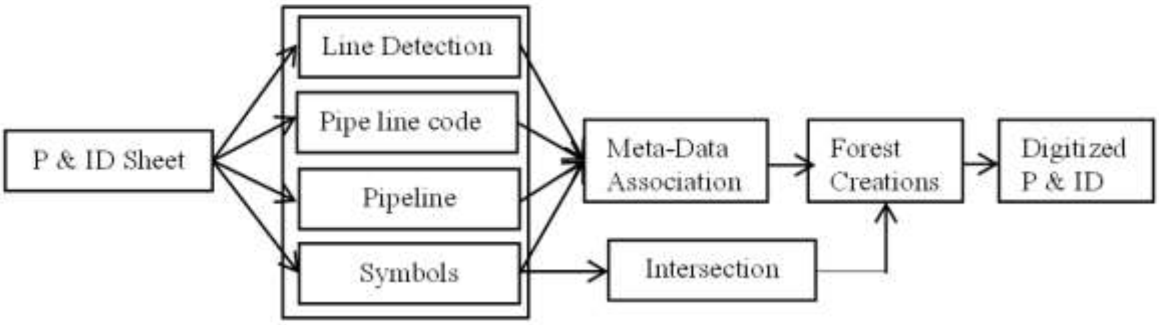

Fig. 1: Proposed pipeline for digitizing engineering diagrams

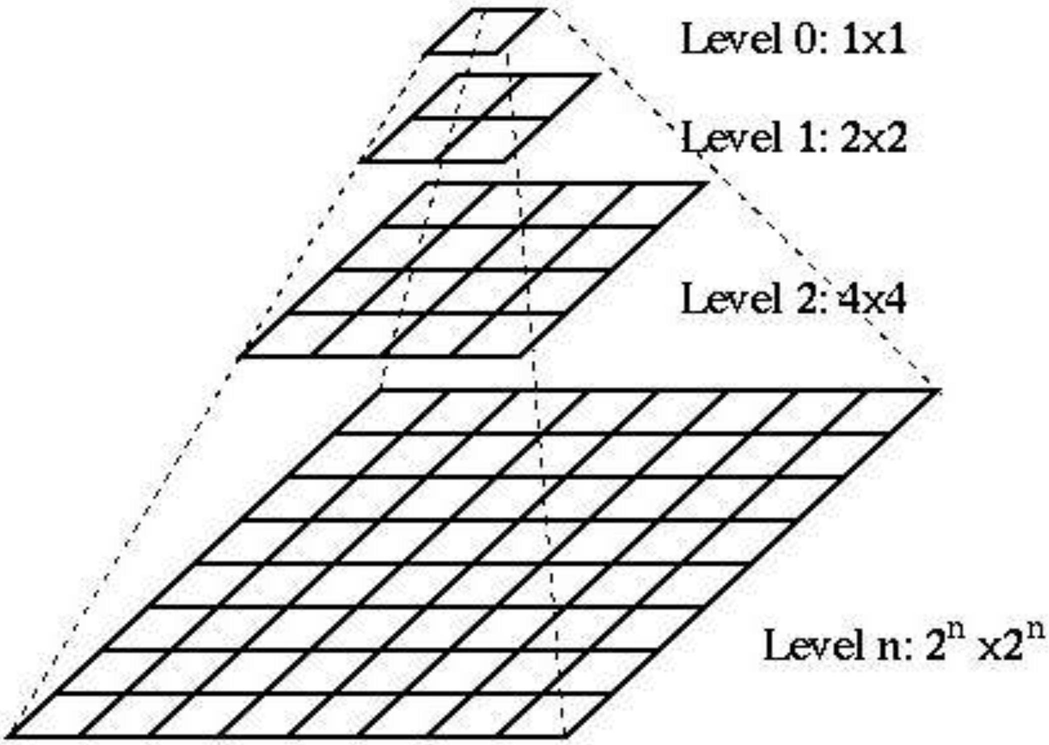

Fig. 2: Image pyramid model

Matching

Once the images to be matched are pre- processed then these images are then rotated at 0, 90, 180, 270 degrees, this is done to maximize the chances of matching along with the scale invariance which is obtained due to creation of the image pyramid. During template matching, the proposed pyramid search algorithm identifies the pairs (template position, template orientation) rather than sole template positions from the input image. Once the template positions are identified, the images from the image pyramid are matched and normalized cross correlation scores calculated as shown in Eq. 1:

$$NCC = \frac{\sum_{x=1}^{i}\sum_{y=1}^{j}\left(T\left(x,y\right)\right).\left(B\left(x,y\right)\right)}{\sqrt{\sum_{x=1}^{i}\sum_{y=1}^{j}\left(T\left(x,y\right)\right)^{2}.\sum_{x=1}^{i}\sum_{y=1}^{j}\left(B\left(x,y\right)\right)^{2}}} \tag{1}$$

In the above given formula for Normalized Cross Correlation, each pixel from the template *T* and the base image *B* are compared. For each comparison the individual pixel product is calculated, which is done in a sliding window method. At each stride, the formula is applied and the product calculated, finally the square roots are calculated and normalization is done using Mean Squares method.

Post Processing

Next, the symbols are recognized and extracted from the scanned P&ID where the detection was done based on the database where the symbols are stored. Post detection the symbols are cut from the image. This is done to reduce the time taken for total computation of symbol detection and reduces the rate of false positives. Since the templates are rotated in all angles of occurrence, the recognitions score increases and those symbols which are identified but not recognized are entered into the database.

Creation of P&ID in Terms of Data

Once all the data is processed, tree search is used to associate the components to each other.

Association Engine

Finally, once all the components are detected, the final association stage begins. Our final step is to associate these components with each other and finally represent the P&ID components in an association: Pipeline Code to Pipeline Association: This is done on a heuristic basis, where the distance between each pipeline and pipe-code tags, is calculated using Euclidean Norm. The pipeline and pipe-code tag having the shortest distance are associated together. Symbol to Pipeline association: Here a database of L2 Norm distance between each detected symbol and pipeline, is maintained. This database is calculated for each component in the engineering diagram and is approx. the size of 200 MB. The symbol is associated with the closest pipe-line.

Experimental Results

Symbols and other equipment were recognized through template matching and 91% accuracy was found. Accuracy is the ratio of total correctly predicted with total predicted cases (Swain *et al*., 2019a-b). Symbols found in the P&ID are registered and recognized. The detector accuracy is calculated by the number of correctly recognized symbols divided by the number of total symbols which exist. It was seen that symbols with similar features such as nozzles and Tesseract OCR (https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/33418.pdf) engine was used to perform text recognition and was 85% accurate. Using the initial language set of the Tesseract was low, but the OCR performance was improved by training on the misrecognized text as show in the figure below. When the symbol overlaps with the text or rotation is present or when the text is long on in length, in these cases the text recognition is low. In order to remove these issues, the symbols are masked before Text recognition is applied. The Table 1 shows the results of recognition results of symbols, lines and text in CAD-converted PDFs and scanned PDFs.

For summarization, recognition validation 91%. As for the symbol recognition, the recognition rates for symbols and equipment in Computer Aided Design-scanned PDF were 92 and 87% for image scanned PDF's. For Line recognition the recognition rates are 91% for scanned pdf and 88% for images and finally text recognition was 88% for Computer Aided Design files and 82% for images. Since the Computer Aided Design PDF files are of higher DPI, these files had a higher recognition rate compared to the images. The recognition rates of text were worse in comparison to symbols and other equipment representations. Albeit best-known OCR engines such as Google Vision (https://cloud.google.com/vision) were adopted to improve recognition accuracy. These were the following most commonly unrecognized elements: (a) Flanges (b) Lines such as horizontal or vertical and separated lines, (c) Overlapped text or text having similar characters due to font types and misread text. Once symbols and text are recognized, line recognition is done. For line recognition, the image is read as a blob. The data which is recognized is stored as an XML file. This is due to the recognized symbols and symbols present in the scanned P&ID are different from each other, symbols are mapped to each other before the conversion, this is done by physically mapping the recognized symbol name and the recognized symbol name in a CSV file.

Table 1: The results of recognizing elements in P&ID's

Target data	Objects	Registered symbols	Total elements in P&ID	Recognized elements	Unrecognized elements	Recognition Rate
CAD-converted Drawings	Symbols	43	426	384	34	90.14%
	Lines	-	350	314	36	89.71%
	Texts	-	1033	918	105	88.86%
Scanned Drawings	Symbols	36	426	400	36	94.00%
	Lines	-	270	252	18	93.33%
	Texts	-	1017	897	40	88.20%

Conclusion and Future Work

This study provided an end-to-end pipeline for digitizing engineering diagrams. This was based on recognition and classification of the document design information by automatic digitization of P&I drawings with a high degree of accuracy in a short period of time. Also, this study provides a method to obtain a digitized P&ID from a scanned file database by recognizing a symbol, line and text from the P&ID. This pipeline recreates an engineering drawing digitally by automating most of the repetitive tasks such as creating the drawings, line listings and instrument cluster listing with a high degree of accuracy in a very small amount of time. This has a direct correlation with engineering productivity by automating tedious tasks Most of the tasks can be automated such as drawing creation, line listings and instrument list calculation with high accuracy in a short period of time. This improves engineering productivity by automating repetitive tasks and drawing is digitized automatically. This also solves the usual issues of time consumption, missing items and misspellings. For further work based on this study, machine learning algorithms based on neural networks would be used to improve the accuracy. Another key part being the fundamental concepts researched for this, especially conversion of engineering diagrams as seen in this study, could be used and extensible to other types of engineering diagrams such as structural diagrams, electrical and instrumentational wiring along with HVAC diagrams and therefore can be further developed.

Acknowledgement

We would like to express our special thanks of gratitude to the management of Pandit Deendayal Energy university, Gandhinagar and Vishwakarma Institute of technology, Pune who gave us the golden opportunity to do this collaborative research work which also helped us in enhancing our knowledge in the field of Artificial Intelligence and image processing.

Author's Contributions

Premanand Ghadekar: Research problem identification and plan preparation.

Shaunak Joshi: Preparing the solution using programming code.

Debabrata Swain: Preparing the solution using programming code.

Biswaranjan Acharya: Data analysis.

Manas Ranjan Pradhan: Manuscript writing.

Pramoda Patro: Literature survey.

Ethics

The published work is purely implemented by the author. There is no conflict of interest in this work.

References

Adam, S., Ogier, J. M., Cariou, C., Mullot, R., Labiche, J., & Gardes, J. (2000). Symbol and character recognition: application to engineering drawings. International Journal on Document Analysis and Recognition, 3(2), 89-101. https://link.springer.com/article/10.1007/s100320000033

Ah-Soon, C. (1997, August). A constraint network for symbol detection in architectural drawings. In International Workshop on Graphics Recognition (pp. 80-90). Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/3-540-64381-8_41

Brown, R. M., Fay, T. H., & Walker, C. L. (1988). Handprinted symbol recognition system. Pattern Recognition, 21(2), 91-118. doi.org/10.1016/0031-3203(88)90017-9

Chiang, J. Y., Tue, S. C., & Leu, Y. C. (1998). A new algorithm for line image vectorization. Pattern recognition, 31(10), 1541-1549. doi.org/10.1016/S0031-3203(97)00157-X

Elyan, E., Moreno-García, C., Jayne, C. (2018) Symbols Classification in Engineering Drawings. In Proceedings of the International Joint Conference on Neural Networks, Rio de Janeiro, Brazil, 8–13 July 2018.

Gellaboina, M. K., & Venkoparao, V. G. (2009, February). Graphic symbol recognition using auto associative neural network model. In 2009 Seventh International Conference on Advances in Pattern Recognition (pp. 297-301). IEEE. https://ieeexplore.ieee.org/abstract/document/4782795

Guo, T., Zhang, H., & Wen, Y. (2012). An improved example-driven symbol recognition approach in engineering drawings. Computers & Graphics, 36(7), 835-845. doi.org/10.1016/j.cag.2012.06.001

Han, C. C., & Fan, K. C. (1994). Skeleton generation of engineering drawings via contour matching. Pattern recognition, 27(2), 261-275. doi.org/10.1016/0031-3203(94)90058-2

Joseph, S. H. (1989). Processing of engineering line drawings for automatic input to CAD. Pattern Recognition, 22(1), 1-11. doi.org/10.1016/0031-3203(89)90032-0

Kacem, A., Belaïd, A., & Ahmed, M. B. (2001). Automatic extraction of printed mathematical formulas using fuzzy logic and propagation of context. International Journal on Document Analysis and Recognition, 4(2), 97-108. https://link.springer.com/article/10.1007/s100320100064

Lu, T., Yang, H., Yang, R., & Cai, S. (2007). Automatic analysis and integration of architectural drawings. International Journal of Document Analysis and Recognition (IJDAR), 9(1), 31-47. https://link.springer.com/article/10.1007%2Fs10032-006-0029-6

Lu, Z. (1998). Detection of text regions from digital engineering drawings. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(4), 431-439. https://ieeexplore.ieee.org/abstract/document/677283

Messmer, B. T., & Bunke, H. (1995, August). Automatic learning and recognition of graphical symbols in engineering drawings. In International Workshop on Graphics Recognition (pp. 123-134). Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/3-540-61226-2_11

Messmer, B. T., & Bunke, H. (1997, September). Fast error-correcting graph isomorphism based on model precompilation. In International Conference on Image Analysis and Processing (pp. 693-700). Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/3-540-63507-6_262

Müller, J., Fregin, A., & Dietmayer, K. (2018, October). Disparity sliding window: object proposals from disparity images. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 5777-5784). IEEE. https://ieeexplore.ieee.org/abstract/document/8593390

Nagasamy, V., & Langrana, N. A. (1990). Engineering drawing processing and vectorization system. Computer Vision, Graphics and Image Processing, 49(3), 379-397. doi.org/10.1016/0734-189X(90)90111-8

Rahman, M. A., Amin, M. F. I., & Hamada, M. (2020, August). Edge Detection Technique by Histogram Processing with Canny Edge Detector. In 202020 3rd IEEE International Conference on Knowledge Innovation and Invention (ICKII) (pp. 128-131). IEEE. https://ieeexplore.ieee.org/abstract/document/9318922

Swain, D., Pani, S. K., & Swain, D. (2019). An efficient system for the prediction of coronary artery disease using dense neural network with hyper parameter tuning. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 8, 6S. https://www.ijitee.org/wp-content/uploads/papers/v8i6s/F61520486S19.pdf

Swain, D., Pani, S., & Swain, D. (2019). Diagnosis of coronary artery disease using 1-D convolutional neural network. Int. J. Recent Technol. Eng.(IJRTE), 8(2). doi.org/10.35940/ijrte.b2693.078219.

Wei, Y., Zhang, Z., Shen, W., Zeng, D., Fang, M., & Zhou, S. (2017). Text detection in scene images based on exhaustive segmentation. Signal Processing: Image Communication, 50, 1-8. doi.org/10.1016/j.image.2016.10.003

Wenyin, L., Zhang, W., & Yan, L. (2007). An interactive example-driven approach to graphics recognition in engineering drawings. International Journal of Document Analysis and Recognition (IJDAR), 9(1), 13-29. https://link.springer.com/article/10.1007/s10032-006-0025-x

Yu, Y., Samal, A., & Seth, S. (1994). Isolating symbols from connection lines in a class of engineering drawings. Pattern recognition, 27(3), 391-404. doi.org/10.1016/0031-3203(94)90116-3