

# Digital interpretation of sensor-equipment diagrams

Carlos Francisco Moreno-García

The Robert Gordon University, Garthdee Road, Aberdeen, UK

**Abstract.** A sensor-equipment diagram is a type of engineering drawing used in the industrial practice that depicts the interconnectivity between a group of sensors and a portion of an Oil & Gas facility. The interpretation of these documents is not a straightforward task even for human experts. Some of the most common limitations are the large size of the drawing, a lack of standard in defining equipment symbols, and a complex and entangled representation of the connectors. This paper presents a system that, given a sensor-equipment diagram and a few impositions by the user, outputs a list with the reading of the content of the sensors and the equipment parts plus their interconnectivity. This work has been developed using open source Python modules and code, and its main purpose is to provide a tool which can help in the collection of labelled samples for a more robust artificial intelligence based solution in the near future.

**Keywords:** Engineering Drawings, Digitisation, Circle Hough Transform, Text Graphics Segmentation, Optical Character Recognition.

A sensor-equipment diagram (SED) is a type of engineering drawing which is commonly used in the Oil & Gas industrial practice to depict how a group of sensors are interconnected to a certain section of an oil rig or a plant. These drawings are composed of a central *main grid* with multiple pieces of equipment, plus a series of circular shapes which represent sensors. An example is shown in Figure 1. Notice that there are two types of sensors: *Local*, which are connected to the main grid through a solid line, and *Panel Mounted*, which are connected to the local sensors through dashed lines.

In recent years, the Oil & Gas industry has shown a particular interest in developing systems that can digitise and interpret a large amount of SEDs in order to migrate printed drawings towards a paperless environment. Nonetheless, experts have realised the difficulty of the automation of this task due to several factors, most notably poor image quality, the large size of a SED, a lack of clarity on the delimitations of an equipment symbol and the complexity of understanding the connectivity. For instance, the SED shown in Figure 1 is an image of  $4460 \times 2544$  *pixels* (approx 1.5 Mb) which contains 164 sensors (not all of them connected to the equipment parts) and four pieces of equipment which can only be identified empirically, since there is no particular standard for differentiating and delimiting equipment parts. Moreover, notice that connectors may overlap or gap each other.

Digitisation and interpretation of engineering drawings from the Oil & Gas industry is not a new problem. In fact, the first literature dates back to the 1980's [5], [8] for a type of drawings called piping and instrumentation diagram (P&ID), which is a much harder type of drawing to digitise, in part because of its larger size, complexity and amount of symbols used. In the 1990's, Howie et al. [7] developed a system which was capable of digitising simple P&IDs in DFX format by using a set of symbols previously loaded by the user.

In 2016 Banerjee et al. [1] presented a system for the automatic linking of construction and manufacturing engineering drawings, where a series of circular symbols called *callouts* are used to establish a link between two pages. To that aim, they implemented a function based on Hough circle detection [3] and some rules to distinguish between true callouts and any other circular shapes in the drawing.

Most recently, Moreno-Garcia et al. presented a heuristics-based method to detect and segregate a series of commonly found shapes in P&IDs. Afterwards, a state of the art digitisation methodology called text/graphics separation [9] was applied to the remaining drawing. This framework showed improved results for text detection. Some of the shapes detected in the P&IDs were continuity labels (i.e. arrow-like shapes), polygons and circular sensors. Later, this methodology was later applied by Elyan et al. [4] to collect a dataset of P&ID symbols and perform classification experiments.

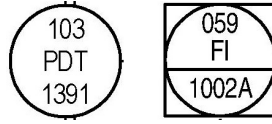
The tool starts by importing the image as a grayscale bitmap. Then, this image is binarised using a standard thresholding algorithm [11]. Afterwards, the system requests the user to perform two types of selections, the area of interest and the equipment part(s) area(s), through the use of a Python-based module called Sloth<sup>1</sup>. The purpose of the manual imposition of the area of interest is twofold; firstly, to discard unwanted elements such as the SED title, margin or other elements considered noise, and secondly, in case that not all of the SED is meant to be digitised. Since there are currently no set of rules that delimit what exactly constitutes each equipment part, the tool requests the users to do an approximate selection of what he or she considers an equipment part. It is expected that, as this tool is used progressively and more labelled examples are obtained, eventually a machine learning algorithm can be trained to automatically detect equipment parts. Figure 2 shows these selections on the example SED.

Once the area of interest is segregated, a circle detection algorithm based on OpenCV<sup>2</sup> Hough circles, which was successfully implemented in previous work for P&IDs [10], is applied to the remaining image. All detected sensors are stored as individual images, as shown in Figure 3. To differentiate between *Local* and *Panel Mounted* sensors, a vertical scan is executed for each sensor image. If one or more horizontal continuous lines are found, then the sensor is flagged as *Panel Mounted*.

---

<sup>1</sup> <http://sloth.readthedocs.io/en/latest/>

<sup>2</sup> <https://opencv.org/>



**Fig. 3.** Example of *Local* (left) and *Panel Mounted* (right) sensor images.

To segregate the text inside the sensors, the largest and outermost contours of each sensor image are discarded, assuming these to be the sensor shape and any noisy pixels surrounding the sensor. In addition, for the case of *Panel Mounted* sensors, an additional detection step for large and elongated contours is implemented. This allows the identification and removal of the horizontal line dividing the text within the sensor. Then, a connected component analysis (CCA) is applied on the remaining image to obtain the height and width of each text character. This will be useful in a later stage to detect the text naming each equipment part. Finally, the sensor text is read using the Pytesseract OCR<sup>3</sup> module.

To automatically detect the text naming each equipment part, a CCA is run on each equipment part area to detect all shapes which are approximately the height and width of an average text character found in sensors. Then, a morphological *brushing* operation [9] is executed, with the aim of filling the gap between text characters and creating contiguous strings of black pixels. The widest stream of contiguous pixels reveals the location of the equipment name, which is then read using Pytesseract OCR.

Finally, the image containing the connectors and the main grid is analysed using a line detection method similar to the one proposed in [9]. This outputs a list of the starting point, endpoint and length of each line. For each local sensor, the system finds the closest line and iteratively checks the line list to find a line segment which "follows the path" (i.e. has a start/endpoint close to the start/end point of another line), until one of the following two conditions are met: 1) the start/endpoint of a line reaches an area marked as an equipment part or 2) there are no further lines that follow the path and an equipment area has not been reached. Once all local sensors are processed, the tool shows the user the final image depicting the local (blue) and panel mounted (red) sensors, the equipment parts names (pink) and the connectors (green). Moreover, the system outputs the list of sensors and the equipments to which these are connected. Both of these outputs are shown in Figure 4.

This paper presents a system which, given an engineering drawing known as a SED, produces a list of the connectivity between the sensors and the equipment parts contained on the main grid. By requesting the user to select the area of interest and the equipment parts, the system automatically finds the sensors, reads the content of the sensors and equipment parts, and deduces the connectivity between these shapes. The method has been successfully showcased

<sup>3</sup> <https://anaconda.org/ijstokes/pytesseract>

to industrial partners of the Oil & Gas sector and has been mounted in one of their servers for testing in future projects<sup>4</sup>.

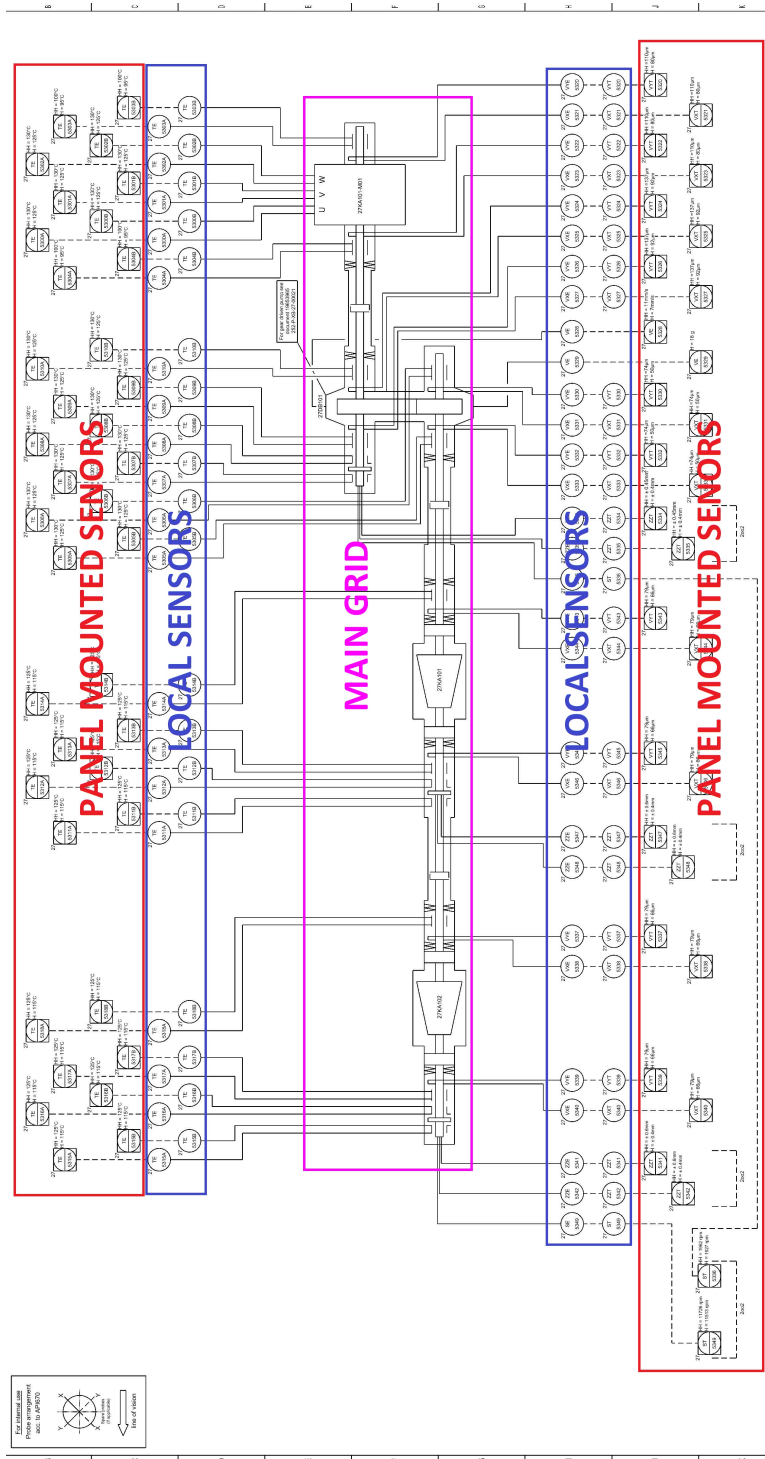
It is important to note that this work opens a niche area of engineering drawing analysis in the light of deep learning advancements. Literature regarding the application of machine learning for the digitisation and interpretation of engineering drawings is still scarce, and most importantly, there is an insufficient amount of labelled data which can aid on training automated systems to perform these tasks. While some work based on neural networks has been presented for the digitisation of other assets such as circuit diagrams [2] and P&IDs [6], these mostly dedicate efforts to the identification of recurrent shapes, but neither consider the presence of rare shapes such as equipment parts, nor the contextualisation of the connectivity between symbols. It is expected that by developing semi-automatic solutions such as the one presented in this paper, it is possible to generate a considerable amount of labelled engineering drawings which eventually can serve as an input for artificial intelligence based solutions.

## References

1. P. Banerjee, S. Choudhary, S. Das, H. Majumdar, R. Roy, and B. B. Chaudhuri. Automatic Hyperlinking of Engineering Drawing Documents. *Proceedings of the 12th IAPR International Workshop on Document Analysis Systems (DAS 2016)*, pages 102–107, 2016.
2. T. Cheng, J. Khan, H. Liu, and D. Yun. A symbol recognition system. In *Proceedings of the Second International Conference on Document Analysis and Recognition - ICDAR'93*, pages 918–921, 1993.
3. R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(April 1971):11–15, 1971.
4. E. Elyan, C. F. Moreno-Garcia, and C. Jayne. Symbols classification in engineering drawings. In *International Joint Conference on Neural Networks (IJCNN)*, 2018.
5. M. Furuta, N. Kase, and S. Emori. Segmentation and recognition of symbols for handwritten piping and instrument diagram. pages 626–629, 1984.
6. M. K. Gellaboina and V. G. Venkoparao. Graphic symbol recognition using auto associative neural network model. In *Proceedings of the 7th International Conference on Advances in Pattern Recognition, ICAPR 2009*, pages 297–301, 2009.
7. C. Howie, J. Kunz, T. Binford, T. Chen, and K. H. Law. Computer interpretation of process and instrumentation drawings. *Advances in Engineering Software*, 29(7-9):563–570, 1998.
8. M. Ishii, Y. Ito, M. Yamamoto, H. Harada, and M. Iwasaki. An automatic recognition system for piping and instrument diagrams. *Systems and computers in Japan*, 20(3):32–46, 1989.
9. Z. Lu. Detection of text regions from digital engineering drawings. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(4):431–439, 1998.
10. C. F. Moreno-García, E. Elyan, and C. Jayne. Heuristics-Based Detection to Improve Text / Graphics Segmentation in Complex Engineering Drawings. In *Engineering Applications of Neural Networks*, volume CCIS 744, pages 87–98, 2017.
11. N. Otsu. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1):62–66, 1979.

---

<sup>4</sup> <http://circuits-dev.azurewebsites.net/>



**Fig. 1.** An example of a sensor-equipment diagram.

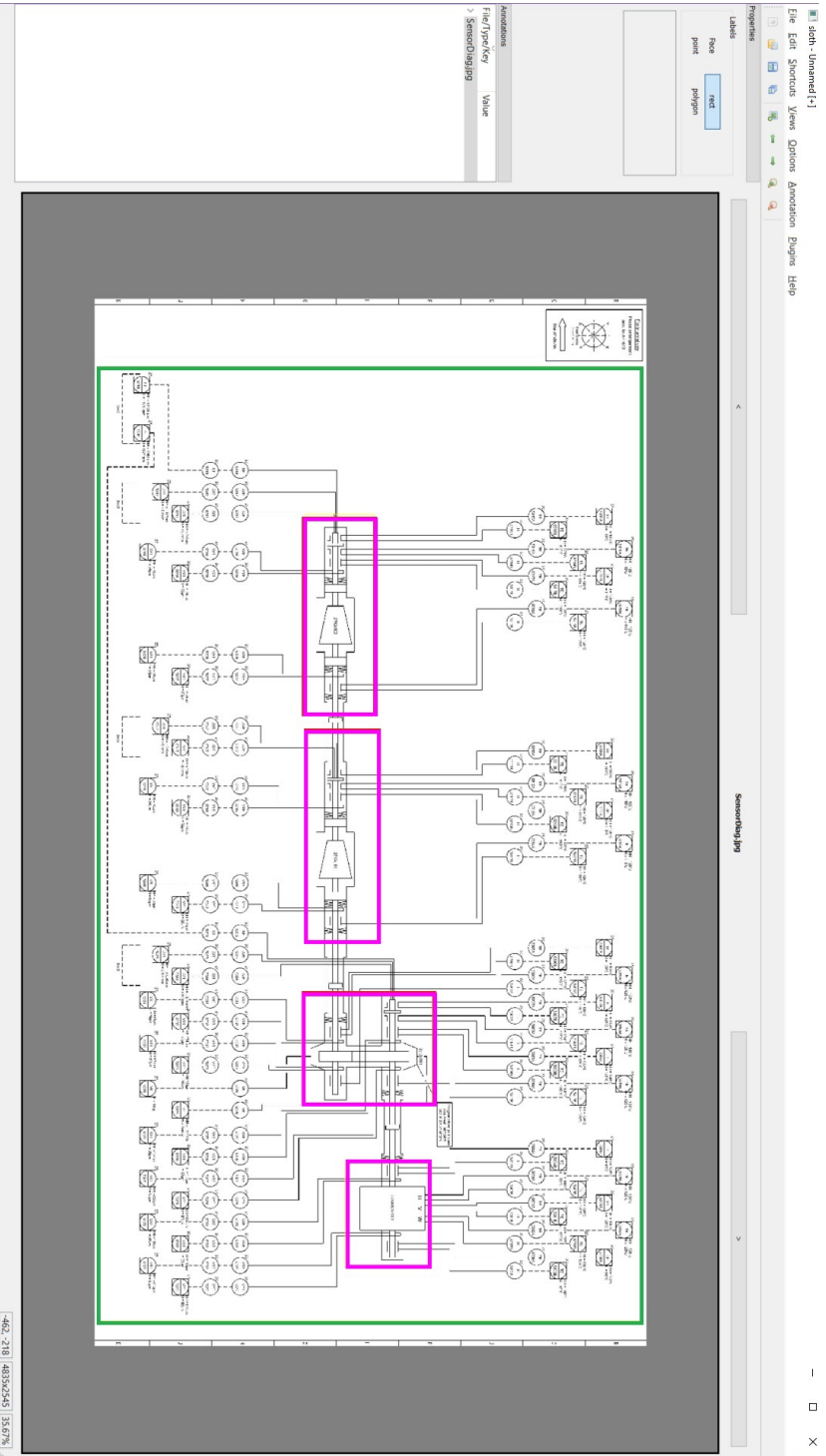
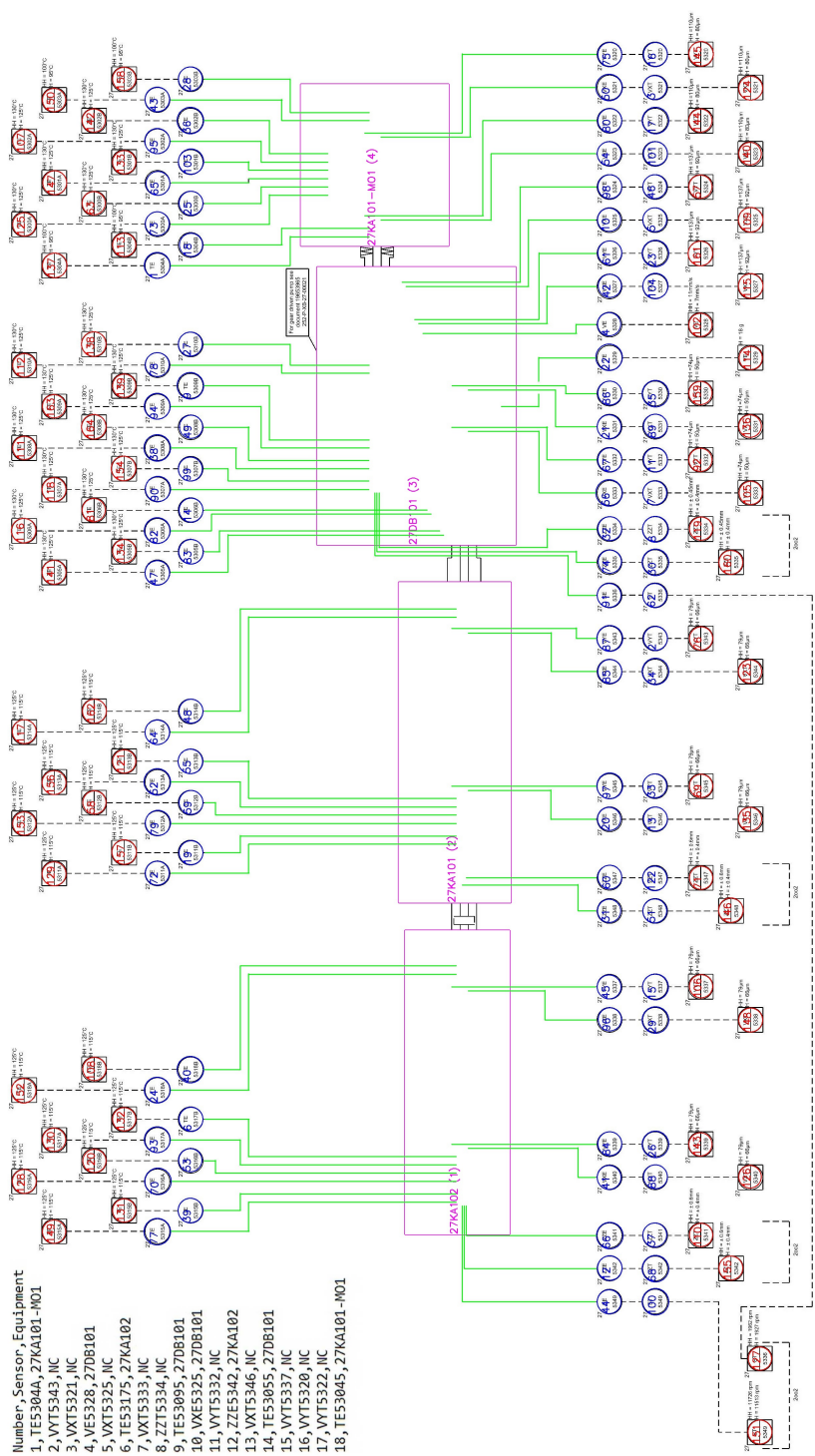


Fig. 2. An example of selecting the drawing area (green) and equipment parts (pink) of a sensor-equipment diagram using Slot.



**Fig. 4.** Output depicting local sensors (blue), panel mounted sensors (red), equipment (pink) and interconnection lines (green), plus a portion of the sensor-equipment connection list is shown