Towards Automatic Digitalization of Railway Engineering Schematics

Stefano Frizzo Stefenon[1,2(✉)], Marco Cristoforetti[1], and Alessandro Cimatti[1]

[1] Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
{sfrizzostefenon,mcristofo,cimatti}@fbk.eu
[2] Department of Mathematics, Computer Science and Physics, University of Udine, Via delle Scienze 206, 33100 Udine, Italy

Abstract. Relay-based Railways Interlocking Systems (RRIS) carry out critical functions to control stations. Despite being based on old and hard-to-maintain electro-mechanical technology, RRIS are still pervasive. A powerful CAD modeling and analysis approach based on symbolic logic has been recently proposed to support the re-engineering of relay diagrams into more maintainable computer-based technologies. However, the legacy engineering drawings that need to be digitized consist of large, hand-drawn diagrams dating back several decades. Manually transforming such diagrams into the format of the CAD tool is labor-intensive and error-prone, effectively a bottleneck in the reverse-engineering process. In this paper, we tackle the problem of automatic digitalization of RRIS schematics into the corresponding CAD format with an integrative Artificial Intelligence approach. Deep learning-based methods, segment detection, and clustering techniques for the automated digitalization of engineering schematics are used to detect and classify the single elements of the diagram. These elementary elements can then be aggregated into more complex objects leveraging the domain ontology. First results of the method's capability of automatically reconstructing the engineering schematics are presented.

Keywords: Computer vision · Deep learning · Engineering drawings

1 Introduction

Engineering Drawing (ED) is derived from descriptive geometry, which is the science that aims to represent three-dimensional objects in the plane (drawing sheet, drawing board, etc.), thus allowing the resolution of infinite problems on the paper plane. For many years, technical EDs have been made by hand, requiring designers to spend a considerable amount of time developing the drawing and editing it. Nowadays, technical drawings for engineering are developed in Computer-Aided Design (CAD) software, where it is possible to make alterations

Supported by Fondazione Bruno Kessler.

R. Basili et al. (Eds.): AIxIA 2023, LNAI 14318, pp. 453–466, 2023.
https://doi.org/10.1007/978-3-031-47546-7_31

easily and quickly. However, the projects that were made before the popularization of CAD software remain archived on paper [1].

When it is necessary to use a not digitized project, a big effort goes into interpreting what has been designed and redesigning the project in CAD. For this reason, many authors have been researching ways to interpret EDs through computer vision methods, and in particular deep learning-based techniques show promise for accomplishing this task [2].

In digitizing EDs through deep learning, several techniques can be applied to improve the model's ability to classify objects in the projects. In this context, preprocessing is performed to highlight the image variations; for this purpose, thresholding techniques can be applied. After preprocessing, shape detection can be performed in specific or holistic categories, features extracted, and the classification process is conducted [3].

According to Günay, Köseoğlu, and Yıldırım [4] Convolutional Neural Networks (CNNs) may be applied to classify drawings made by hand, their research shows that CNNs perform well in the classification of electrical circuit drawings. One of the major challenges in identifying electrical components in hand-drawn drawings is the great difference between designs since there is no linear pattern of hand-drawing development [5]. Classification accuracy using CNNs can achieve acceptable results in this task applied to EDs.

In the railway signaling domain, the Railway Interlocking Systems (RIS) control the movement of trains, allowing or denying their routing according to safety rules. Relay diagrams are a commonly used abstraction for modeling relay-based RIS, describing such systems by graph-like schematics that show the connections between the electrical components. However, verifying these diagrams for safety is challenging due to their complexity and the lack of tools for automatic verification [6].

Several EDs of the RIS remain on paper because these systems were designed before the popularization of CAD. With models based on CNN becoming increasingly faster and capable of classifying images with high accuracy, the interpretation of ED of RIS becomes relevant research. From a model that can interpret the drawings, transform this information into matrices, and verify the semantics, it will be possible to have a tool that will help update these projects. So far, the application of machine learning related to railways is based on the analysis of RIS automation or improved timetable analysis [7], having room and need for ED evaluation using deep learning techniques, as presented here.

The proposed method presented in this paper has a combination of several techniques, its overall workflow is: Initially, the You Only Look Once (YOLO) [8] Deep Neural Network is applied for object detection, and based on the found objects a new drawing without them is used to focus especially on the segments. To identify the segments, the Probabilistic Hough Transform (PHT) [9] combined with the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [10] is applied. Based on the segments found a graph is built to evaluate the electrical circuits of the EDs.

The contributions of this work to the digitization of EDs for RIS are summarized in the following:

– developing an automatic method for digitizing engineering schematics. The proposed approach is capable of redrawing EDs that are printout and is faster than the manual labor required to render these EDs in executable machines.
– improving segment detection using the Density-based Spatial Clustering of Applications with Noise in the Probabilistic Hough Transform. Using a clustering method, it is possible to group overwritten segments, improving the quality of the digitalization of these elements.
– proposing a graph problem based on the electrical connections of the RIS. Building on the ontological characterization of the domain, this paper proposes a graph-based problem to aggregate the basic elements into more complex objects.

The remainder of this paper is organized as follows: Sect. 2 brings a literature review on works related to EDs. Section 3 presents the applied methodology. Section 4 brings the evaluation of the proposed approach. Finally, Sect. 5 presents a conclusion and discusses future works.

2 Related Works

Technical drawings have a large application in industrial processes because they provide important information about the structure where the project is applied, besides interconnections between equipment. EDs are used in different industries, such as oil and gas, mechanical, construction, and other fields of engineering [11]. Nowadays, digitizing these designs is becoming extremely important, and to update the projects, it is necessary to have them digitized.

In industrial projects, the specific identification of components is difficult to achieve since there is a similarity between the standard graphic symbols used. According to Yun et al. [12], to perform the object identification process in EDs, it is necessary to have a structure composed of several steps: region proposal evaluation, feature extraction, and classification. From the results of this evaluation, it is possible to classify the proposed regions and extract the symbolic information by means of dummy detection.

To recognize and extract important information from engineering diagrams, it is necessary to perform a preprocessing stage in which the alignment of the drawing, the removal of outer edges, and the title box withdrawal are performed [13]. This information can make it difficult to identify objects and, at this stage does not contribute to the classification since, in the initial analysis, the object is not related to the place where it is installed.

The relationship between the location of the object and the place where it belongs is important as soon as there is global knowledge of the project, so it is possible to make a validation of what has been identified. Mani et al. [14] presented a strategy using graph search to traverse a diagram through its lines and discover interconnected symbols.

An essential aspect to be considered is that CNNs may perform well for object classification; however, they may have difficulty identifying text characters, lines, and tables. For this reason, to improve the interpretability of EDs, it is promising to use combined models, in which part of the network is used specifically for character detection, making the solution more powerful [15]. Deep learning-based models can be used especially for classification, making the network more accurate for this task.

In the work of Kim et al. [16] the interpretation of EDs is conducted using high-resolution images through specialized models for symbol detection and text recognition after preprocessing the images. The selection of symbols into labels makes object identification a clearer task to be performed. The Generalized Focal Loss (GFL) and feature selective anchor-free methods can show slightly better results for localizing objects in projects compared to other CNNs. Using different backbones applied to GFL can further improve the results in symbol identification, highlighting the Residual neural Network (ResNet) using 50 layers has resulted in an accuracy of 0.970 for this task.

Specialized models such as the Character Region Awareness For Text detection (CRAFT) proposed by Baek et al. [17] and the Convolutional Recurrent Neural Network (CRNN) presented by Shi, Bai, and Yao [18] stand out for text recognition. With this, Kim et al. [16] showed that to have a better performance in the analysis of EDs, it is required to use specific techniques for text detection, such as CRAFT and CRNN, and other techniques for object identification, such as GFL. Even with a large number of classes, this approach results in a complete and robust solution for interpreting EDs.

Li, Yuhui, and Xiaoting [19] developed a CNN-based analysis for classifying EDs using three categories: mechanical EDs, text drawings, and electrical EDs. In their study, argumentation techniques were applied to increase the size of the dataset by rotating, slicing, and including random noise. This practice is common due to the need for a large database to train the CNN model. The results showed that it is possible to have an accuracy of up to 0.987 for the classification of different project types. For their application, the use of four convolutional layers resulted in higher accuracy than using three.

An important issue to take into account in ED classification is the large number of different classes with a little variation and imbalance that makes this even more challenging. In the research of Elyan, Jamieson, and Ali-Gombe [20], a combined approach is proposed to deal with all these challenges. In the proposed method, bounding box detection is initially performed to locate and recognize symbols. A deep generative adversarial neural network is used to deal with class imbalance. From this approach, it is possible to train the network with a small number of images and achieve highly accurate results.

To deal with the problem of the small database faced in this type of analysis, Bickel, Schleich, and Wartzack [21] presented an approach to augment the dataset based on randomly creating symbols and illustrations. From this, a deep learning network can be used to test the dataset and recognize the symbols in the EDs using more images. Since there are a large number of classes to be classified, a promising strategy in the first stage is to group similar circuit components, and in the next stage, the circuits belonging to the same group can be classified using CNN-based models. According to Dey et al. [22], using a 2-stage structure with 20 classes of circuit components, a classification accuracy of 0.973 can be achieved, which is higher than single-stage models.

Since Relay-based RIS (RRIS) contain thousands of combinations of component instances, the interpretation of ED manually is a hard task. In this context, deep computer vision is an outstanding solution. Since besides object detection it is also needed to detect segments, this paper proposes a method that combines techniques to achieve this goal, which is innovative for this application, the proposed method is explained in the next section.

3 Proposed Method

The method proposed in this paper combines deep learning-based models with probabilistic methods and clustering techniques. To meet the project goal, the eighth generation of the You Only Look Once (YOLOv8) is applied for object detection, and the PHT combined with the (DBSCAN) are employed for segment detection.

3.1 Data Preparation

The EDs of the Italian railway company (*Rete Ferroviaria Italiana* - RFI) have high-resolution, reaching more than 30,000 pixels wide by 4,000 pixels high, with images having different electrical connections from different parts of the railway stations. The first step in this project, in order to be able to apply machine learning models, was to make standardized cutouts of the engineering drawings. The 640, 1280, and 2560-pixel cutouts were considered, as in the examples in Fig. 1.

Since drawings have different sizes (height and width), with horizontal dimensions larger than vertical dimensions it is necessary to standardize the images to be considered. Therefore, it was standardized to use 640-pixel cutouts, and with this, the dataset for training and validation of the model was created and organized. Given this size variation, the number of cutouts depends on each project's original size.

The cutouts are disjoint from the complete ED, and then for the training the symbols that were not wholly presented were not considered. Since not complete symbols were disregarded the model was trained based on identifying only whole symbols as needed for the reconstruction of the EDs. After identifying all symbols and labels, a reconstruction of the entire drawing is performed and objects in the borders are manually evaluated.

Training the model using images only of each symbol (from a pallet) did not have effective results in the testing phase (inference). For this reason, no dataset augmentation was used. In the initial experiments, including augmented data reduced the model's ability to be effective for real applications. Since the best

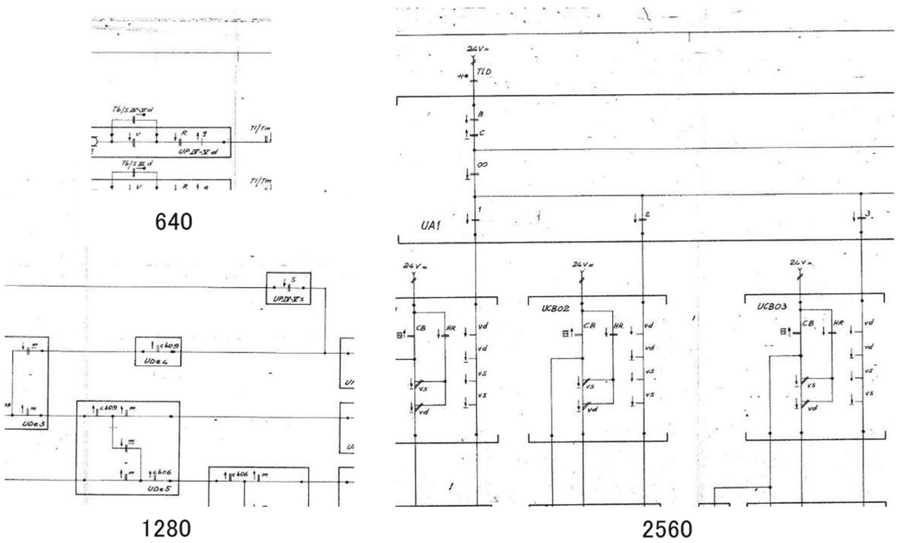

Fig. 1. Engineering drawing of a relay-based railway interlocking system.

results were obtained using real images, only this type of image was used to improve the applicability of the proposed model.

Symbols are the main objects to be identified in the project since electrical connections occur between them. The labels and specifiers are additional information that is attached to the symbols. The specifiers, labels, and connection points (dots) were annotated to obtain a complete dataset. The total classes considered in the project are 133: letters [0 to 9, a to z, A to Z], dots [0 to 11 variations], specifiers [arrows up or down], and symbols [C00 to C56]. Regarding the letters, the Tesseract Optical Character Recognition (OCR) [23] was used as a baseline comparison to the YOLO results.

3.2 Object Detection Method

The YOLO is a single-shot algorithm, meaning the detection and classification are performed in a single run. The model has been improved over the years, and the latest version released by Ultralytics [24] is the eighth generation (YOLOv8).

The images under consideration are divided into a $S \times S$ grid, where each grid square predicts the object's bounding box, corresponding to its degree of confidentiality [25]. Thus, the confidence of classes (cl) of objects (obj), is denoted by:

$$pr(cl_i \mid obj) \cdot pr(obj) \cdot IoU_{pred}^{truth} = pr(cl_i) \cdot IoU_{pred}^{truth}. \tag{1}$$

where i is the respective class under evaluation, considering the intersection over union (IoU) of the predicted ($pred$) bounding boxes compared to the ground truth ($truth$) [26]. In this paper, 133 classes are considered. Once the image is divided into grids, a class probability map is computed to identify the target objects and bounding boxes to determine if the desired objects are situated in this confidence region.

3.3 Segment Detection to Create Connections

The Hough Transform (HT) is employed to determine the parameters of features as lines in an image [27]. In EDs of the RFI, the lines (segments) are used to connect electrical components (symbols) and create a logic for the control of the RIS. It should be noted that the symbols have rules to be connected; thus, there is a need to identify the segments properly. Therefore evaluations of the system logic can be performed to improve the safety of the RIS.

To apply the HT, a binary image is used as input, where each active pixel is part of an edge feature, and the HT maps every pixel in the Hough space. The Canny edge detector was applied to have the input image binarization [28]. For line detection, a single edge pixel is mapped to a sinusoid in parameter space (θ, p), representing all possible lines that could pass through the point in the image. If several points are collinear, their sinusoids in parameter space will intersect. Finding the points in parameter space where most of the sinusoids intersect provides the parameters for the lines in the input image. The process is referred to as the search stage [29].

The standard HT returns lines that cross the entire image, then the PHT is used to have the start and end points of the segments. The PHT is defined as the logarithm of the probability density function of the output parameters, taking into account all input features [30].

Let's consider an input image with a set X_n of feature $x_1, x_2, ..., x_n$ and a specified point in the parameter space y. The probability density function in Hough space $H(y)$ is $p(y \mid X_n)$ and then the PHT is denoted by:

$$H(y) = \ln[p(y \mid X_n)] \tag{2}$$

which, by the Bayes' rule, is

$$H(y) = \sum_{i=1}^{n} \ln[p(x_i \mid y)] + \ln[p(y)] + C \tag{3}$$

where the probability distribution $p(y)$ is considered uniform and C is the arbitrary constant. The PHT result may have short segments instead of a complete segment since preprocessing using canny edge detection is required before computing the PHT. Therefore, the detection may be a set of small segments that can be nearby and overwritten, based on which clustering of the segments must be performed.

In this paper, DBSCAN is applied to connect segments that are close apart or overwritten. This method is a non-parametric density-based clustering algorithm. Taking a set of points in a given space, it clusters the close points, scoring as outliers the points that are isolated in low-density regions [31]. Therefore, combining DBSCAN with PHT is promising for reconstructing the detected segments. The complete architecture applied here is presented in Fig. 2.

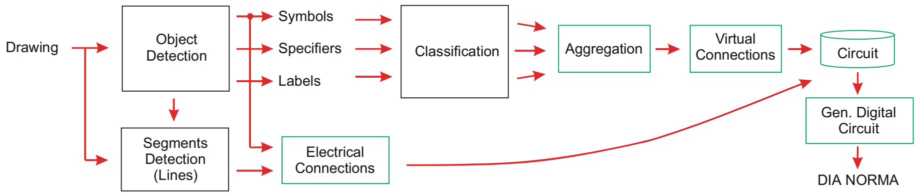

Fig. 2. Architecture of the proposed method.

4 Results and Discussion

In this section, the results of the application of the proposed method will be presented. An analysis will be performed regarding the use of a CNN, focusing on the detection of labels, electrical connections, specifiers, and symbols. The detection of labels will be compared to the well-established Nanonets and Tesseract OCR. From the identification of the objects under evaluation, the results of segment detection, clustering, and the final outcome of the automatically reconstructed drawing will be presented.

4.1 Experiment Setup

The proposed approach is based on the PyTorch library using Python programming language. The experiments were conducted on a cluster, with a Graphics Processing Unit (GPU) NVIDIA Tesla V100 and 32 GB of random-access memory. To maximize processing time, the experiments of random initialization were run in parallel using five GPUs, with the same requirements to be computed. The performance evaluation measures are the standard for this kind of task: precision, recall, F-measure, and mean Average Precision (mAP).

4.2 Object Detection and Classification

The YOLOv8 (nano) is evaluated considering all considered 133 classes being these labels (letters), electrical connections (dots), specifiers, and symbols. For the training of the models, five RFI projects were considered, containing 850 cutouts of 640 pixels in which 2,323 symbols are shown having their respective labels and specifiers.

The results presented here are relative to the model validation scores, since in the test phase using inference images the used metric is confidence in the predictions given a threshold. For a complete assessment, the maximum (max), minimum (min), mean, median (med.), and standard deviation (std dev.) of the performance measures are computed. In Table 1 the influence of starting weights is presented considering 50 experiments with different random seeds.

The major goal of this model is to detect the objects to continue with image processing, for this reason, the average of the mAP@[0.5] equal to 0.80276 is an acceptable value. There was a difference of 4.32% comparing the highest

Table 1. Statistical analysis considering different seeds.

Measure	Max	Min	Mean	Med.	Std dev.
Precision	0.83	0.72	0.78	0.79	2.82×10^{-2}
Recall	0.81	0.74	0.77	0.77	1.89×10^{-2}
F1-score	0.81	0.74	0.78	0.77	1.37×10^{-2}
mAP@[0.5]	0.82	0.78	0.80	0.80	1.01×10^{-2}
mAP@[0.5:0.95]	0.39	0.34	0.38	0.38	8.09×10^{-3}

and lowest mAP values. This result shows that there may be some variation in the model due to network initialization, however, this is acceptable for object detection needed at this stage.

To avoid the influence of the data selection, the k-fold cross-validation using 5-folds was used. The results of this evaluation are presented in Table 2. An observation to be made is that when images with higher levels of noise are used for validation, the training time becomes considerably longer.

Table 2. Evaluation of cross-validation.

Measure	Max	Min	Mean	Med.	Std dev.
Precision	0.78	0.69	0.74	0.76	3.95×10^{-2}
Recall	0.80	0.57	0.72	0.73	7.60×10^{-2}
F1-score	0.79	0.63	0.73	0.75	5.59×10^{-2}
mAP@[0.5]	0.82	0.64	0.75	0.78	6.29×10^{-2}
mAP@[0.5:0.95]	0.47	0.36	0.42	0.42	3.64×10^{-2}

The utilization of noisy data to perform the model validation resulted in an average reduction of mAP@[0.5] of 4.93%. Although this is a high reduction in mAP, this variation is given by the characteristics of the data used. Even with high noise data, the model is still capable of achieving an average mAP@[0.5] equal to 0.75349.

The output of the model is a text file that has the class of the found object, the bounding box that determines the geometric position of the object, and the prediction confidence of the detection. The resulting text file is used as a reference for segment detection since the objects identified in this phase should not be considered when segment detection is performed.

4.3 Evaluation of Standard OCR Methods

For comparison, an experiment is conducted using Tesseract and Nanonets for character recognition on EDs of the RFI. The preliminary results showed that because these standard models were trained for text identification based on standard fonts, character identification using these techniques was inefficient.

Using the Tesseract engine, the results were under 40% for both precision and recall, by using the Nanonets, the results were under 55% for these same metrics. It was observed that these methods were suitable only for identifying characters in the project legends. Based on these results, the YOLO is defined for label recognition, where all characters are new classes.

4.4 Results of Segment Detection

To detect the segments properly, the symbols, specifiers, electrical connections, and labels are removed. In this procedure, the output of the YOLO is used as a reference to define the locations that should not be considered segments because they are objects. A new image is then created in which the bounding boxes of the identified objects are redrawn blank and are disregarded in the segment detection. Then, the PHT is applied as presented in Fig. 3A.

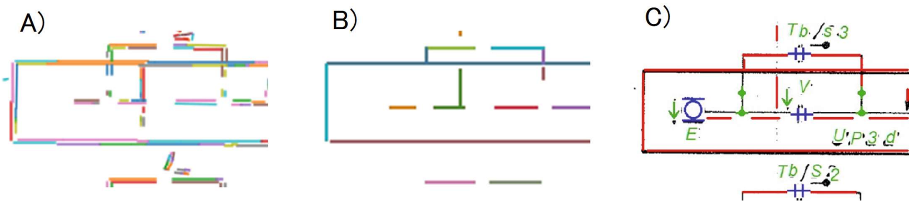

Fig. 3. Probabilistic Hough Transform applied in the ED of the RFI: A) original PHT, B) DBSCAN applied to the PHT, C) reconstructed ED.

The use of PHT results in the detection of multiple segments, instead of one complete segment, i.e. instead of identifying a connection between two symbols the method overwrites segments. For this reason, DBSCAN was applied, whereby segments that have a short distance between them and are in the same direction (vertical or horizontal) are clustered. The result of the application of this technique in the PHT is presented in Fig. 3B.

Using this clustering approach to join smaller segments, the proposed method had acceptable results and can be applied even to other engineering problems. After defining the new segments, the drawing can be reconstructed, considering the position of the identified objects beyond the segments. The results of this combination are presented in Fig. 3C.

The misclassification in object detection is not a problem for segment detection considering that only the position of the bounding box is used to remove the objects and apply the PHT and DBSCAN. Missing objects can be a problem given that the PHT would recognize them. To solve this issue only horizontal and vertical segments are considered with a maximum variation angle of 5°.

4.5 Electrical Connection Graph

To improve the meaning of the connection of the identified objects, logical reasoning rules can be applied to validate the connection of the symbols. Since symbols are the focus of electrical connections, specifiers and labels can be considered as additional information. To build the graph of electrical connections, Identifier Numbers (IDs) are defined for each identified object and segment, as presented in Fig. 4.

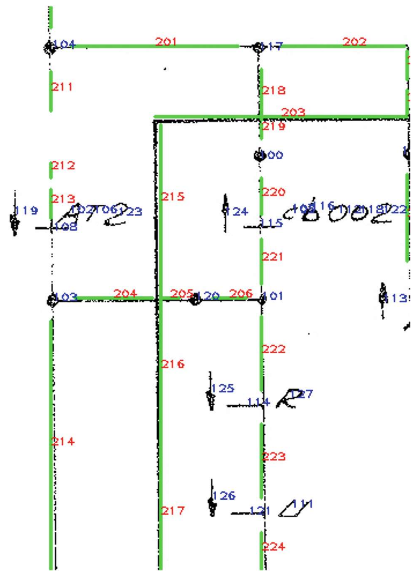

Fig. 4. Drawing with IDs of segments and objects. (Color figure online)

For standardization, the object IDs are shown in blue with values between 100 and 199, and the segment IDs are shown in red and have values between 200 and 299. Due to the low sharpness of some segments of the schematics, these may not be identified using this method, such as the segment below the symbol ID 108 (Fig. 4). This characteristic is related to the available dataset and makes the task even more challenging.

In general, the method is able to identify in an acceptable way the segments and objects under consideration, especially the symbols which are the focus of the analysis. Based on the proximity of the segment to the edges of the bounding boxes of the symbols and the electrical connections (points in the drawing) a graph is created, as shown in Fig. 5.

In the presented example of the resulting graph all the connections were identified, this occurred mainly because, in the distance calculation, the edges of the bounding boxes were considered, preliminary evaluations using the center of the symbols resulted in lower accuracy. In this example, the detected graph meets the expectations in comparison with the ground truth.

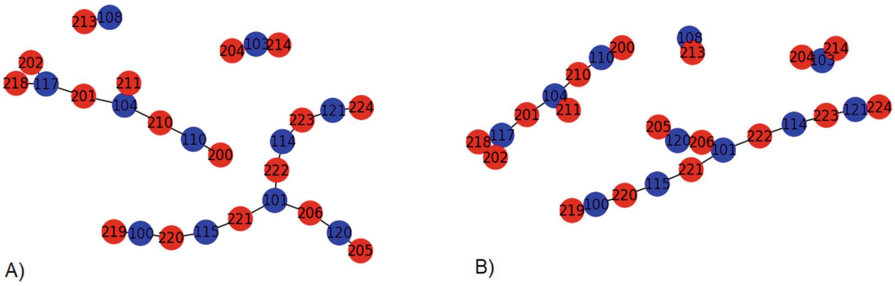

Fig. 5. Electrical connection graph: A) ground truth, B) detected connections.

5 Conclusion and Final Remarks

The digitization of engineering drawings is a necessity in many fields since many projects come from a legacy in which they were originally hand-designed. Identifying text is a challenge because handwritten drawings do not follow a font standardization, making this challenge even more difficult. Established OCR techniques are not enough to solve this task, therefore it is necessary to train specific models for the identification of each character.

The proposed method successfully had the ability to identify text, symbols, specifiers, and segments in EDs. With a mean F1-score of 0.77513, and a mean mAP@[0.5] of 0.80276, considering 50 experiments, the YOLOv8 had acceptable results to be applied in the field. Especially for segment detection, the segmentation and clustering techniques were combined, resulting in a promising approach to automate the redrawing EDs.

These results highlight that the redraw of ED based on deep learning methods is flexible. The presented method returns the symbols and label positions with acceptable results, being a functional approach that represents the original EDs. Given the importance of accuracy in reconstructing EDs of the RRIS, the verification and approval of an expert is still necessary, therefore the proposed method aims to speed up the manual task of redrawing from scratch.

In future work, the electrical connections between symbols can be used to strengthen the specification of electrical components. Furthermore, using graphs, a more adequate determination of labels and specifiers can be achieved since their use in the projects follows a pattern, and using logical reasoning graphs can improve the redrawing of the projects, besides being a tool for verification and validation of what is being presented.

References

1. Rica, E., Moreno-García, C.F., Álvarez, S., Serratosa, F.: Reducing human effort in engineering drawing validation. Comput. Ind. **117**, 103198 (2020). https://doi.org/10.1016/j.compind.2020.103198
2. Mizanur Rahman, S., Bayer, J., Dengel, A.: Graph-based object detection enhancement for symbolic engineering drawings. In: Barney Smith, E.H., Pal, U. (eds.)

ICDAR 2021. LNCS, vol. 12916, pp. 74–90. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86198-8_6

3. Moreno-García, C.F., Elyan, E., Jayne, C.: New trends on digitisation of complex engineering drawings. Neural Comput. Appl. **31**(6), 1695–1712 (2019). https://doi.org/10.1007/s00521-018-3583-1
4. Günay, M., Köseoğlu, M., Yldrm, ŏ.: Classification of hand-drawn basic circuit components using convolutional neural networks. In: 2020 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA), Ankara, Turkey, vol. 1, pp. 1–5. IEEE (2020). https://doi.org/10.1109/HORA49412.2020.9152866
5. Günay, M., Köseoğlu, M.: Classification of hand-drawn circuit components by considering the analysis of current methods. In: 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Istanbul, Turkey, vol. 4, pp. 1–5. IEEE (2020). https://doi.org/10.1109/ISMSIT50672.2020.9255047
6. Huang, L.: The past, present and future of railway interlocking system. In: International Conference on Intelligent Transportation Engineering (ICITE), Beijing, China, vol. 5, pp. 170–174. IEEE (2020). https://doi.org/10.1109/ICITE50838.2020.9231438
7. Huang, P., Li, Z., Wen, C., Lessan, J., Corman, F., Fu, L.: Modeling train timetables as images: a cost-sensitive deep learning framework for delay propagation pattern recognition. Expert Syst. Appl. **177**, 114996 (2021). https://doi.org/10.1016/j.eswa.2021.114996
8. Stefenon, S.F., Singh, G., Souza, B.J., Freire, R.Z., Yow, K.C.: Optimized hybrid YOLOu-Quasi-ProtoPNet for insulators classification. IET Gener. Transm. Distrib. **17**(15), 3501–3511 (2023). https://doi.org/10.1049/gtd2.12886
9. Ahmad, R., Naz, S., Razzak, I.: Efficient skew detection and correction in scanned document images through clustering of probabilistic Hough transforms. Pattern Recogn. Lett. **152**, 93–99 (2021). https://doi.org/10.1016/j.patrec.2021.09.014
10. Chen, Y., Zhou, L., Bouguila, N., Wang, C., Chen, Y., Du, J.: BLOCK-DBSCAN: fast clustering for large scale data. Pattern Recogn. **109**, 107624 (2021). https://doi.org/10.1016/j.patcog.2020.107624
11. Elyan, E., Garcia, C.M., Jayne, C.: Symbols classification in engineering drawings. In: International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, vol. 1, pp. 1–8. IEEE (2018). https://doi.org/10.1109/IJCNN.2018.8489087
12. Yun, D.Y., Seo, S.K., Zahid, U., Lee, C.J.: Deep neural network for automatic image recognition of engineering diagrams. Appl. Sci. **10**(11), 4005 (2020). https://doi.org/10.3390/app10114005
13. Yu, E.S., Cha, J.M., Lee, T., Kim, J., Mun, D.: Features recognition from piping and instrumentation diagrams in image format using a deep learning network. Energies **12**(23), 4425 (2019). https://doi.org/10.3390/en12234425
14. Mani, S., Haddad, M.A., Constantini, D., Douhard, W., Li, Q., Poirier, L.: Automatic digitization of engineering diagrams using deep learning and graph search. In: Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, USA, pp. 673–679. IEEE (2020). https://doi.org/10.1109/CVPRW50498.2020.00096
15. Kang, S.O., Lee, E.B., Baek, H.K.: A digitization and conversion tool for imaged drawings to intelligent piping and instrumentation diagrams. Energies **12**(13), 2593 (2019). https://doi.org/10.3390/en12132593
16. Kim, H., et al.: Deep-learning-based recognition of symbols and texts at an industrially applicable level from images of high-density piping and instrumentation

diagrams. Expert Syst. Appl. **183**, 115337 (2021). https://doi.org/10.1016/j.eswa.2021.115337
17. Baek, Y., Lee, B., Han, D., Yun, S., Lee, H.: Character region awareness for text detection. In: Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, USA, pp. 9357–9366. IEEE (2019). https://doi.org/10.1109/CVPR.2019.00959
18. Shi, B., Bai, X., Yao, C.: An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **39**(11), 2298–2304 (2017). https://doi.org/10.1109/TPAMI.2016.2646371
19. Li, L., Yuhui, C., Xiaoting, L.: Engineering drawing recognition model with convolutional neural network. In: International Conference on Robotics, Intelligent Control and Artificial Intelligence, pp. 112–116. Association for Computing Machinery, New York (2019). https://doi.org/10.1145/3366194.3366213
20. Elyan, E., Jamieson, L., Ali-Gombe, A.: Deep learning for symbols detection and classification in engineering drawings. Neural Netw. **129**, 91–102 (2020). https://doi.org/10.1016/j.neunet.2020.05.025
21. Bickel, S., Schleich, B., Wartzack, S.: Detection and classification of symbols in principle sketches using deep learning. Proc. Des. Soc. **1**, 1183–1192 (2021). https://doi.org/10.1017/pds.2021.118
22. Dey, M., et al.: A two-stage CNN-based hand-drawn electrical and electronic circuit component recognition system. Neural Comput. Appl. **33**, 13367–13390 (2021). https://doi.org/10.1007/s00521-021-05964-1
23. Kumar Garai, S., Paul, O., Dey, U., Ghoshal, S., Biswas, N., Mondal, S.: A novel method for image to text extraction using tesseract-OCR. Am. J. Electron. Commun. **3**(2), 8–11 (2022). https://doi.org/10.15864/ajec.3202
24. Ultralytics: YOLOv8 in PyTorch. https://github.com/ultralytics/ultralytics. Accessed 4 Aug 2023
25. Souza, B.J., Stefenon, S.F., Singh, G., Freire, R.Z.: Hybrid-YOLO for classification of insulators defects in transmission lines based on UAV. Int. J. Electr. Power Energy Syst. **148**, 108982 (2023). https://doi.org/10.1016/j.ijepes.2023.108982
26. Singh, G., Stefenon, S.F., Yow, K.C.: Interpretable visual transmission lines inspections using pseudo-prototypical part network. Mach. Vis. Appl. **34**(3), 41 (2023). https://doi.org/10.1007/s00138-023-01390-6
27. El Hajjouji, I., Mars, S., Asrih, Z., El Mourabit, A.: A novel FPGA implementation of Hough Transform for straight lane detection. Eng. Sci. Technol. Int. J. **23**(2), 274–280 (2020). https://doi.org/10.1016/j.jestch.2019.05.008
28. Lu, Y., Duanmu, L., Zhai, Z.J., Wang, Z.: Application and improvement of canny edge-detection algorithm for exterior wall hollowing detection using infrared thermal images. Energy Build. **274**, 112421 (2022). https://doi.org/10.1016/j.enbuild.2022.112421
29. Zhao, K., Han, Q., Zhang, C.B., Xu, J., Cheng, M.M.: Deep Hough transform for semantic line detection. IEEE Trans. Pattern Anal. Mach. Intell. **44**(9), 4793–4806 (2022). https://doi.org/10.1109/TPAMI.2021.3077129
30. Wang, L., Yang, Y., Shi, J.: Measurement of harvesting width of intelligent combine harvester by improved probabilistic Hough transform algorithm. Measurement **151**, 107130 (2020). https://doi.org/10.1016/j.measurement.2019.107130
31. Li, S.S.: An improved DBSCAN algorithm based on the neighbor similarity and fast nearest neighbor query. IEEE Access **8**, 47468–47476 (2020). https://doi.org/10.1109/ACCESS.2020.2972034