A research project for extracting object-oriented descriptions of piping and instrumentation diagrams

ESTEBAN ARROYO, STEVE ROYSTON, ALEXANDER FAY, MARIO HOERNICKE, PABLO RODRIGUEZ – Documentation in process industries can rise to bewildering dimensions. Furthermore, the difficulties of accessing it can be compounded by the range in formats used, including traditional paper and elementary digital representations. When a plant owner decides to consolidate this information – for example, to prepare a modernization process – bringing this eclectic documentation into a single, accessible and up-to-date format can require a gargantuan effort. ABB has explored means of efficiently extracting models from engineering documents in an automated and consistent manner. A joint research project by ABB and the Helmut Schmidt University (HSU) in Hamburg, Germany, resulted in a method based on optical recognition and semantic analysis that converts piping and instrumentation diagrams (P&IDs) into object-oriented models.
The sheer size and complexity of process-industry plants combined with the numerous modifications and additions effected throughout their life-cycles results in vast amounts of legacy documentation. This includes structural models, functional descriptions, P&IDs and equipment specifications. Although these documents embody a rich source of information that should be exploited in planning and operational activities, this is currently hindered by the challenges entailed by their timely localization and data extraction [1].

In current industrial practice, existing engineering documentation is digitized manually by scanning, storing and indexing. The resulting files contain unstructured information that cannot be fully exploited for automation tasks as their underlying content does not take the form of a data model. Addressing this, ABB has, in cooperation with HSU, devised methods for the extraction of object-oriented (OO) representations from existing design documents, specifically P&IDs [2].

Previous research has shown that the use of P&IDs can support different tasks along the plant’s life-cycle such as automated HAZOP (hazard and operability) studies [3], detection of design patterns [4], derivation of simulation models [5] and fault detection and diagnosis [6]. Existing approaches are all based on the prerequisite that P&IDs must be described in OO formats (eg, IEC 62424 CAEX/AutomationML [7,8] or ISO 15926 [9]). Unfortunately, this is not typically the case in existing process facilities as most P&IDs are filed in non-computer-interpretable forms.

Input documents are composed of four fundamental artifacts, namely parametric forms, nonparametric forms, text annotations, and underlying semantic content.

From pixels to meaning
The method extracts OO models from mixed text-graphics documents, concretely P&IDs → 1, recognizing text, symbolic forms and connection topologies. The approach relies on the fact that input documents are composed of four fundamental artifacts, namely parametric forms, nonparametric forms, text annotations and underlying semantic content → 2.

The conversion process assumes that P&IDs are stored in pdf files (pdf/bitmap or pdf/SVG) from which raster images can be extracted. In some cases, however, preliminary process steps are required so that a raster image can be subsequently obtained, particularly when dealing with legacy papers or CAD files → 3. From here, the method executes three procedures for the generation of the computer-interpretable model, namely:

- Optical recognition: OSR and OCR → 4 for the identification of symbols and text annotations.
- Semantic analysis: Interpretation of the functional connotation of graphical forms for the enhancement of the recognition procedure and the improvement of expression capabilities for the model description.
- Representation and verification: Description of captured information as an OO model followed by visual inspection.

Optical recognition (OR)
OR deploys sound image processing techniques for the identification of geometric forms and text identifiers. The process is executed through the recursive application of two methods: OSR and OCR.

Optical symbol recognition
As the first step of OSR, nonparametric symbols (ie, multi-curve forms) such as “vessel E100” and “valve V300” shown in → 1 are localized and matched against predefined libraries of templates. A library
can represent, for instance, plant asset catalogs or sets of symbols commonly used to embody specific devices or processes within P&IDs. The definition of libraries allows for not only modularity, which in turn results in more efficient search procedures, but also effective capture and reuse of structural knowledge within projects. The matching criterion employed is based on the structural features of the examined forms. Three key geometric properties are supported to guarantee an exhaustive search of possible matches – namely, rotation, scaling, and occlusion.

Scores quantifying the extent of resemblance between identified candidates and templates are used to define levels of confidence within the recognition method. Successfully matched objects are suppressed from the image to facilitate the recognition of further artifacts.

In a second step, parametric symbols (i.e., mathematically described forms such as circles or lines), eg, "pipe Pi002" and "sensor L101" in Figure 1, are identified. Recognition methods used within this procedure are capable of identifying possible candidates regardless of their size and orientation. Thresholds can be set to define the minimum dimensions of parametric forms and thereby avoid the misinterpretation of characters as symbols. As with the non-parametric symbols, the reliability of the recognition process can be adjusted based on calculated matching scores. Again, the position coordinates of objects found are suppressed to ease the identification of further artifacts.

Optical character recognition
Based on the notion that the text identifier of an object is located in its surroundings or within the object, the coordinates of identified forms are used to generate search windows. The size of a window is typically proportional to the dimensions of the analyzed image. It can also be defined by the user. An OCR algorithm searches for text within the defined window. If more than one tag is found, nomenclature rules (e.g., regular expressions) are used to discard erroneous strings and a unique tag is assigned to each object. Ambiguous identifications (e.g., due to touching characters) are referred to the user for manual resolution.

Semantic analysis (SA)
In the context of the developed method, SA refers to the incorporation of domain-specific knowledge, particularly functional and structural content, within the recognition of graphical forms and their interrelations. Among other purposes, SA is exploited for connectivity detection and enhancing modeling expression capabilities through the definition of specific interface types.
Beyond connectivity detection, further rules are tested so that mistaken detections can be automatically corrected and/or the user warned.

**Boosting connectivity detection**

Characteristic connections within engineering diagrams are based on polyline arrangements. Hence, connectivity detection requires as a first step the unification of contiguous lines to single linking artifacts. The proximity of such artifacts to structural forms is analyzed and the underlying connectivity of the diagram derived. Semantic rules are applied to enforce the consistency of links. A semantic rule might state, for instance, that a pipeline can be connected to at most two other elements, one at each termination. Where potential instances violate such statements, further rules are tested so that mistaken detections can be automatically corrected and/or the user can be warned.

**Definition of interface types**

The generation of detailed OO models requires distinguishing between different types of connectivity. For instance, the semantics (function and physics) of a vessel-sensor connection is intrinsically different to that of a vessel-pipe link. The former transmits information and is typically implemented by wires, while the latter carries material and is commonly realized through flow couplings.

**Intermediate structural description**

In order to provide a first representation of gathered information, the connectivity, position and dimensions of the elements found are stored respectively in a connectivity matrix and a table of coordinates.

The connectivity matrix has as first row and column the list of recognized elements. These are referred to by their class and tag identifier, eg, vessel E100. The inner entries of the matrix represent the connectivity between elements (row, column). Connection types are indicated by distinctive names: F for material flow and I for information flow.
From paper to digital
Making the most of documentation

The improved exploitation of existing documentation and automated transfer to digital plant models enables increased efficiency of (re)engineering tasks in greenfield and brownfield projects. It can be applied in a number of industries, including chemical, pharmaceutical, oil and gas, water, and power generation.

Intermediate structural descriptions can be codified as spreadsheets (e.g., using Microsoft Excel or data tables in C# Windows Forms) and recursively accessed for the derivation of the targeted OO models. Additionally, they can be used for basic tasks such as generation of part lists or solving queries about the presence of a specific item or item type.

OO Modeling
An OO model, specifically a CAEX/AutomationML model compliant with IEC 62424 [7,8], is derived from the collected data. The procedure starts by creating an object for every entry within the first column of the connectivity matrix. Subsequently, interfaces are added. For example, for vessel E100, two interfaces are created – one of type MaterialFlow (F) and one of type InformationFlow (I). The process concludes by assigning attributes (e.g., position and dimensions). Thus an OO model describing the connectivity and semantic information of the P&ID is obtained.

Visual inspection
Finally, a graphical depiction of the inferred information is generated. The representation overlays the original black and white P&ID with colored forms representing the identified objects. Users can visualize these objects and compare them with the original diagram, effecting changes where necessary. Corrections carried out in the graphical interface have a direct effect on the OO model, thus avoiding the need for additional engineering tools. Previously generated warnings can be used to prioritize the cross-checking procedure.

References