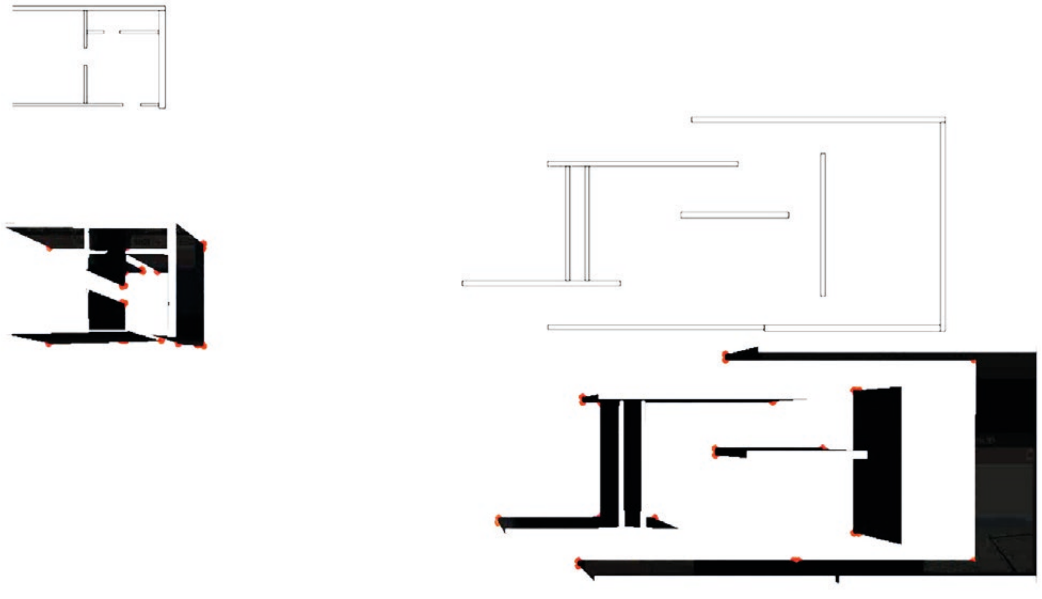
Automatic Recognition Through Deep Learning of Standard Forms in Executive Projects

Devid Campagnolo
Paolo Borin

Abstract

In this paper is presented a possible methodology for automation through the use of deep learning of BIM modeling starting from different types of formats, such as digital processing of paper documents and CAD formats. The work is configured as a proof of concept of a possible contribution that a technique currently scarcely used in the architectural field such as deep learning can bring to the design, in particular in the realization of the information model, which today represents one of the most consuming–time activities.

Keywords
deep learning, computer vision, BIM, automation



doi.org/10.3280/oa-686.57

notBIM to BIM

The production of information, in the construction sector and in a very widespread domain, is undergoing a transformation process in recent years that involves the transition from the use of CAD to the implementation of BIM as regards the generation and extraction of graphic and non–graphic data. This process, although strongly favored by national legislation, is still in a phase in which CAD software often plays a role of primary importance, especially in the initial stages of design, while BIM is sometimes relegated to the final stages of the design process, during the final and executive project, only to be abandoned in the construction phase. It is also true for many realities that a large part of the built heritage preserves documentation in paper or CAD format.

In the production of projects and graphics, the current state of a BIM model shows drawings drawn up in different formats, with different methods and purposes for the design disciplines: there are cases in which there is a need to create a model for the management of an existing building artifact, to draw up a building restoration project, for the variation of methodology that can take place between levels of subsequent technical study in the design, such as the transition from a final project developed in CAD to an executive project developed in BIM.

The creation of a BIM model finds in the modeling of geometric data one of the activities of greatest time consumption, with consequent dilatations in the planning times, which are reflected in an increase in costs for organizations [Deutsch 2011]. The existence of graphic data created with paper or CAD formats, although used as an initial data for BIM modeling, only partially exploits the amount of time spent in the production of such data, with consequent waste of the work carried out previously, as the data starting graph must be manually recreated within the BIM model, like for example the production of digital text files starting from handwritten documents. In addition to being extremely laborious, a data recreation process of this type also depends on the experience and expertise of the operator involved [Koutamanis, Mitossi 1992].

In this work we try to obtain through Deep Learning methodology those geometric data, present in paper or CAD drawings, useful for the creation of BIM objects, and to use such data for the automatic creation of such objects through algorithms. With reference to the existing bibliography, it is a question of updating experiments already carried out with advanced algorithms, evaluating the speed of execution, accuracy and integrability. It is possible to find basic algorithms starting from the early 90's [Kaneko 1992]. Such algorithms mainly work on symbol recognition [Mokhtarian, Abbasi 2004]. These first specialized works, which served as cadastral plans to demonstrate the correctness of the awards, were followed by applications dedicated to architectural representation for the transition from 2D drawings to three–dimensional models [Lewis, Séquin 1998]. The analysis models have become more specialized, in a direction of increasing effectiveness towards different disciplines and electronic media [Dosch et al. 2000; Lu et al. 2005; Yin, Wonka and Razdan 2009]. The application to the generation of BIM models as output has instead been explored only recently in which the recognition is applied to all building elements, such as doors and windows [Lim, Janssen and Stouffs 2018]. In fact, this work aims to show how it is possible to integrate, in a single BIM modeling environment, through visual programming, Computer Vision and Deep Learning libraries.

Methodology

The project exploits existing libraries for Deep Learning and Computer Vision in order to recognize objects from suitably digitized paper formats and to recreate these objects in a BIM environment using VPL. For this purpose, the visual programming environment Autodesk Dynamo in its version DynamoCoreRuntime 2.7.0 was used, given the impossibility of integrating Python libraries for Deep Learning in the current versions integrated into the BIM modeling software Autodesk Revit (Autodesk Dynamo Revit 2.5.0 for Autodesk Revit 2021), and OpenCV (Open Source Computer Vision Library), a software library for Computer Vision and Machine Learning.

The graphic drawings taken into consideration are the plan views, as they represent in most cases the most important documents in the architectural field [Koutamanis, Mitossi 1993].
The process can be divided into 4 stages:

- preparation of the image;
- image processing;
- subdivision of detected objects:
- creation of objects from extracted geometric information.

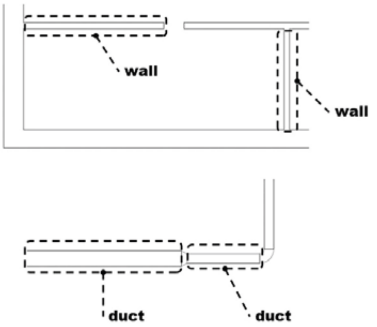


Fig. 1. Element recognition scheme. Elaboration by the authors.

The image preparation phase involves the digitization of images from paper format or the transformation of vector files into matrix form. The search for the most suitable method for the CAD–matrix transformation is currently underway, however the export of tables in JPEG form seems to be the quickest and easiest method for preparing the CAD data.
In the case of digitized paper documents, the creation of tracing elements of the beginning and end of the analysis area and identification of the coordinates, as well as identifying the scale of the objects contained in the document, is deemed necessary.
In the processing phase, the image imported into the Visual Programming Language (VPL) is provided as input to a Python script that uses openCV in order to identify simple geometric shapes formed by 4 vertices in the image. The library uses appropriate filters to identify, through matrix operations, the presence in the image of vertical lines, horizontal lines or edges in order to determine the presence of vertices in a certain region of the image.
The currently implemented algorithm applies a single transformation to grayscale images that provides for the application of a threshold to each pixel, for which the pixel values above the threshold are brought to 0 while the higher values are taken to the maximum value supplied as input, in this case 255. A second function identifies possible outlines of elements in the image, which are subsequently discretized by obtaining the vertices of the figure.
The list of vertices is filtered by selecting only the figures whose number of vertices is equal to 4. The vertices are then decomposed into their own X–Y components and supplied as output.

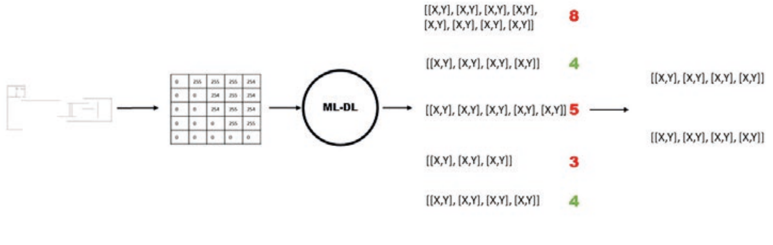

0 255 255 255 254
0 0 254 255 254
0 0 254 255 254
0 0 0 255 255
0 0 0 0 0
ML-DL
[[X,Y], [X,Y], [X,Y], [X,Y], [X,Y], [X,Y], [X,Y], [X,Y]] 8
[[X,Y], [X,Y], [X,Y], [X,Y]] 4
[[X,Y], [X,Y], [X,Y], [X,Y], [X,Y]] 5
[[X,Y], [X,Y], [X,Y]] 3
[[X,Y], [X,Y], [X,Y], [X,Y]] 4
[[X,Y], [X,Y], [X,Y], [X,Y]]
[[X,Y], [X,Y], [X,Y], [X,Y]]

Fig. 2. Algorithm operation diagram. Elaboration by the authors.

The list of coordinates obtained is then used to create the related points in the VPL environment, which will create four–sided polygons that are analyzed in their geometric information in order to perform a further skimming on the objects in the initial image.

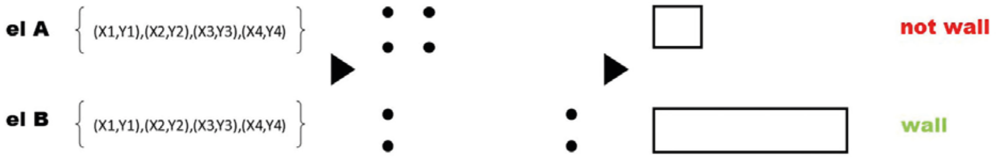

el A
(X1,Y1),(X2,Y2),(X3,Y3),(X4,Y4)
not wall
el B
(X1,Y1),(X2,Y2),(X3,Y3),(X4,Y4)
wall

Fig. 3. Results classification. Elaboration by the authors.

The objects are created starting from the geometric or textual information obtained from the drawing. In the case of textual information, an algorithm can be set up which, using openCV, recognizes text elements, obtains their position and content and associates them with the forms detected using the algorithm described above.

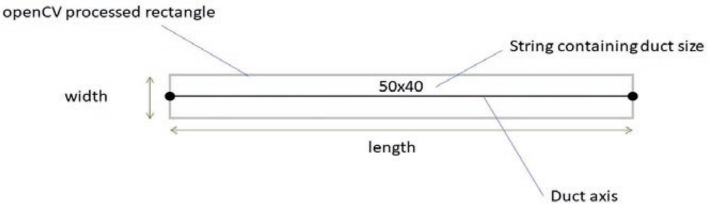


Fig. 4. Information extraction. Elaboration by the authors.

The currently developed algorithm is an extremely simplified resolution of the identification problem and represents a proof of concept of the possibility of integrating this type of operations in a single environment.

For a real application, the algorithm requires further developments such as:

- a part of better pre–processing of the image, in order to ensure better results for non–standardized images from the point of view of color, contrast;
- the training of a neural network that can autonomously classify elements within the image based on pre–training on typical representations of architectural elements in plan (walls, pillars, floors, doors, windows ...).

From this perspective, the algorithm would not be limited to the recognition of rectangular shapes only, but would be aimed at the recognition of elements of any shape and their classification on the basis of the classes provided during the training phase.The algorithm would then obtain, depending on the class, different information for the creation of objects in the BIM environment. For example, for the creation of wall elements, the axis of the wall (major axis of the rectangle) and the thickness (minor axis) would be derived from the identified rectangle, for the creation of door elements, the centroid of the identified polygon (position of the door in the closest identified wall element. In this sense, we would approach an almost complete recognition of the graphic information contained in the original work, which can be divided into three types [Koutamanis, Mitossi 1993]:

- Recognition of the geometric element: extraction of geometric information by element from the graphic;
- Recognition of the Building Element: extraction of classes by element from the graphic;
- Recognition of the spatial articulation, training a neural network to recognize the spaces formed by the association of different elements (concerning the architectural discipline), or zones (concerning the MEP discipline).

Results

The results obtained in the processing of an image representing part of an HVAC system are shown below. Note how an extremely simple form such as that of the rectangle takes on a multitude of meanings in architectural language based on the context in which this graphic symbol is placed.The variation in meaning occurs both in relation to the discipline and in the relationship between the element and the context. A rectangle can therefore represent a wall, a floor or a piece of furniture in the architectural discipline, a pillar or a beam in the structural discipline, a rectangular or circular duct or a connection in aeraulic systems. The need to train a neural network for the discernment of such cases is therefore recognized.

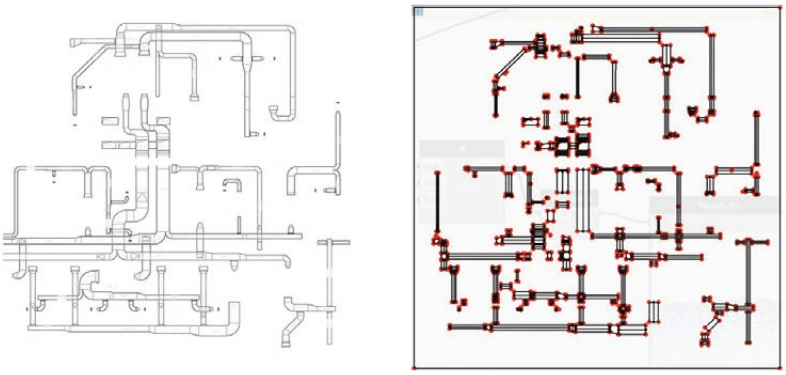


Fig. 5. Information extraction. Elaboration by the authors.

It is possible to note that a sketched and very simplified version of the algorithm already achieves good results in recognizing simple elements and tracing their axis. In this case, a subsequent algorithm will be responsible for the grouping of shapes on the basis of the connection by means of objects of different shapes, representing in this case the fittings of the ducts. At this point you have all the geometric information necessary for the creation of the elements in the BIM environment. The work carried out has undoubtedly demonstrated the greater speed of execution of a deep learning/object recognition algorithm compared to methods of analysis according to geometric formulas, previously experimented by the authors. To this must be added the high accuracy rate in object recognition for simple geometric shapes or scarcely variable shapes achieved by the current available algorithms. However, it is necessary to create more complex neural networks that can recognize and classify a multitude of objects belonging to the common architectural language for a complete or semi–complete automated modeling of BIM objects starting from the considered formats. In order to have a fully operational IT product, it is also necessary to develop a method for verifying the dimensional correctness of the outpute and the subsequent correctness of positioning of the elements in space.

References

Deutsch Randy (2011). *BIM and Integrated Design. Strategies for Architectural Practice*. Hoboken: John Wiley & Sons.

Dosch Philippe et al. (2000). A complete system for the analysis of architectural drawings. In *International Journal on Document Analysis and Recognition.* 3 (2), pp. 102-116.

Kaneko Toru (1992). Line structure extraction from line–drawing images. In *Pattern Recognition*. 25 (9), pp. 963-973.

Koutamanis Alexander (1993). The future of visual design representations in architecture. In *Automation in Construction*, 2, pp. 47-56.

Koutamanis Alexander, Mitossi Vicky (1992). Automated recognition of architectural drawings. In *11th IAPR International Conference of Pattern Recognition*. 1, pp. 660-663.

Koutamanis Alexander, Mitossi Vicky (1998). Spatial representations as the basis of formal and functional analysis. In *CUMINCAD,* pp. 1-10.

Koutamanis Alexander, Mitossi Vicky (1993). Computer vision in architectural design. In *Design Studies* 14. 1, pp. 40-57.

Lewis Rick, Séquin Carlo (1998). Generation of 3D building models from 2D architectural plans. In *CAD Computer Aided Design*. 30 (10), pp. 765-779.

Lim Joie, Janssen Patrick, Stouffs Rudi (2018). Automated generation of BIM models from 2D CAD drawings. In *CAADRIA 2018 – International Conference on Computer–Aided Architectural Design Research in Asia: Learning, Prototyping and Adapting*, pp. 61-70.

Lu Tong, Tai Chiew–Lan, Bao Li, Su Feng, Cai Shijie(2005). A new recognition model for electronic architectural drawings. In *CAD Computer Aided Design*. 37 (10), pp. 1053-1069.

Mokhtarian Farzin, Abbasi Sadegh (2004). Matching shapes with self–intersections: Application to leaf classification. In *IEEE Transactions on Image Processing*. 13 (5), pp. 653-661.

Scheer David Ross (2014). *The Death of Drawing. Architecture in the Age of Simulation*. New York: Routledge.

Yin Xuetao, Wonka Peter, Razdan Anshuman (2009). Generating 3D Building Models from Architectural Drawings: A Survey. In *IEEE Computer Graphics and Applications*. 29 (1), pp. 20-30.

Authors

Devid Campagnolo, Dept. of Civil, Environmental and Architectural Engineering, University of Padua, devid.campagnolo@unipd.it
Paolo Borin, Dept. of Civil, Environmental and Architectural Engineering, University of Padua, paolo.borin@unipd.it

Isbn 9788835125280