Re-Engineering Based on Construction Drawings
- From Ground Floor Plan to Product Model -

V. Berkhahn, C. Kinkeldey, Institute of Computer Science in Civil Engineering, University of Hannover, Germany
(berkhahn@bauinf.uni-hannover.de, kinkeldey@bauinf.uni-hannover.de)
M. Schleinkofer, E. Rank, Lehrstuhl für Bauinformatik, Technische Universität München, Germany
(schleinkofer@bv.tum.de, rank@bv.tum.de)

Summary

For the management or reorganisation of existing buildings, data concerning dimensions and construction are necessary. Often these data are given exclusively by paper-based drawings and no digital data such as a computer based product model or even a CAD-model are available. In order to perform mass calculation, damage mapping or a recalculation of the structure these drawings of the building under consideration have to be analysed manually by the engineer. This is a very time-consuming job. In order to close this gap between drawings of an existing building and a digital product model an approach is presented in this paper to digitise a drawing, to build up geometric and topologic models and to recognise construction parts of the building. Finally all recognised parts are transformed into a three-dimensional geometric model which provides all necessary geometric information for the product model. During this import process the semantics of a ground floor plan has to be converted into a 3D-model.

1 Introduction

Although today most CAD systems offer 3D-modelling techniques, an efficient integration into the planning process is only possible, if in addition to purely geometric features further product-related attributes can be modelled and stored. In principle, all persons involved in a model-based planning process are able to retrieve all necessary information, to process the relevant data and to store the results of their special planning steps back to the model. Several commercial product model systems are characterised by a rather specific and proprietary development. In contrast to this, the International Alliance for Interoperability aims to create the Industry Foundation Classes (IFC 2003) as a generally accepted product model (Neuberg et al. 2002). The development of product models and their integration into the design and planning process is one of the most important and promising topics of Computer Science in Civil Engineering. Within this context the priority program 1103 “network-based co-operative planning processes in structural engineering” (DFG 2004), promoted by the German Research Foundation since 2000, has an outstanding importance.

Whereas model-based planning is now proving its first merits in the construction of *new* buildings, often not even the basic geometric descriptions being necessary for management or reorganisation of *existing* buildings are *digitally* available. In general, these data are given exclusively by paper-based drawings. CAD-systems deliver features to digitised drawings by using digitisation tablets or background images for the drawing area. Significant points and lines are selected manually and the corresponding coordinates are stored in a digital 2D-drawing model. This is a very time consuming job and the semantic interpretation of drawing objects has to be performed by the engineer or draughtsman.

Software tools (e.g. Dosch et al. 2000) have been developed for the graphical analysis and pattern recognition specialised for application in the field of architecture. In the field of constructional engineering these tools do not fulfil practical requirements, because the shape, scale and orientation of drawing objects may widely vary and therefore the methods of pattern recognition are not suitable. Therefore an approach via line identification is chosen for this paper. Different line identification methods such as "orthogonal zig-zag" (Chai et al. 1992), "burning algorithm" (Lindquist und Lee 1996) or "spare pixel tracking" (Yoo et al. 1998) are used in commercial vectorisation systems. The burning algorithm is basic for several line and structural identification tools (Berkhahn 2004) (Schleinkofer 2003). In this paper a method (Berkhahn & Esch 2003) based on medial axes (Wolter 2001) combined with the burning algorithm is used to digitise the pixel-based lines within a drawing.

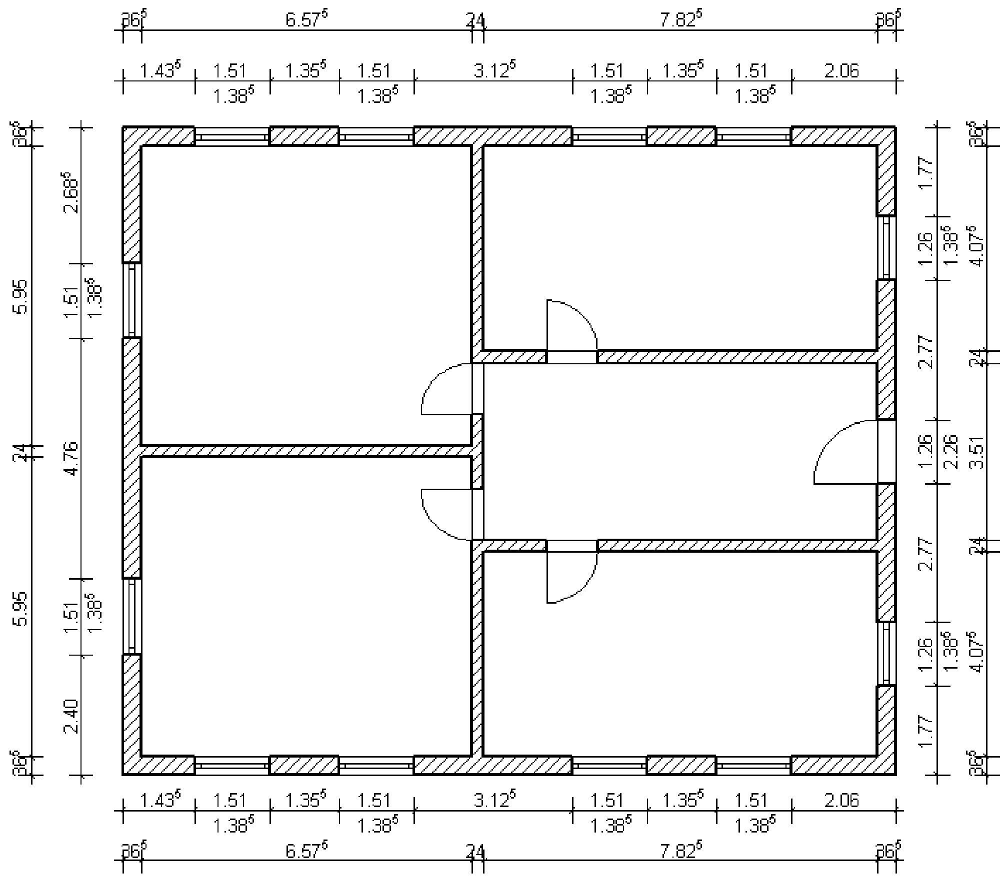

Figure 1: Ground floor plan of a small office building

Based on a topological analysis of all digitised lines a recognition process for construction parts is performed afterwards. In order to explain the presented approach an example of a small office building is chosen (Figure 1). For the detailed explanation of all steps of line identification and construction part recognition a small detail (Figure 2) of the ground floor plan is selected. Finally, all construction parts of the ground floor plan are displayed as a 3D-model within the product model (Figure 12).

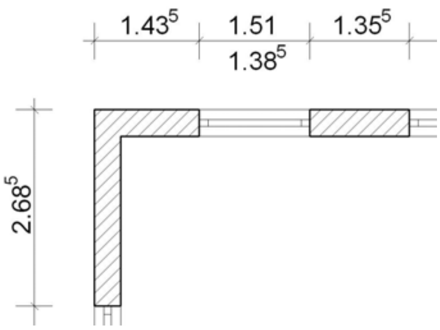

Figure 2: Detail of a ground floor plan

2 Identification of lines in a pixel-based drawing

In the first step of the re-engineering process the construction drawing is scanned and transformed into a pixel-based image. This image consists of a set of pixels ordered in a $n \times m$ grid. Within this grid every pixel p_{ij} can be identified by two independent indices i and j:

$$P = \{p_{ij} \mid 0 \le i \le n; 0 \le j \le m\} \tag{1}$$

Every pixel p_{ij} has different grey scale values $g(p_{ij})$. For the identification of lines the grey scale range of all pixels is separated into a range of background pixels and a range of line pixels. Figure 3 shows the background pixels and line pixels for small detail of a ground floor plan.

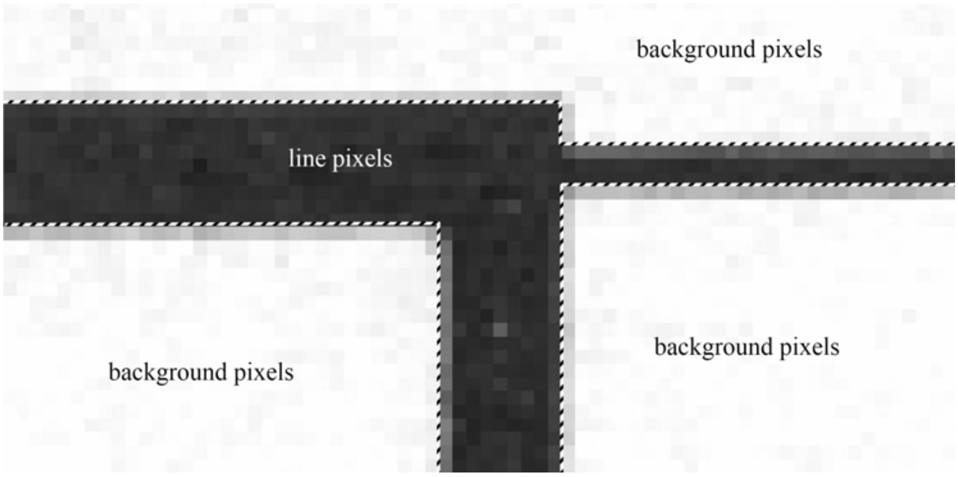

Figure 3: Background and line pixels of a drawing detail

The sets of line pixels and background pixel are subsets of all pixels P of a drawing

$$\begin{aligned} P_{line} &= \{p_{ij} \mid g(p_{ij}) \ge g^* \wedge 0 \le i \le n \wedge 0 \le j \le m\} \\ P_{back} &= \{p_{ij} \mid g(p_{ij}) < g^* \wedge 0 \le i \le n \wedge 0 \le j \le m\} \end{aligned}, \tag{2}$$

where g^* denotes the limit grey scale value, separating line pixels and background pixels. This limit grey scale value has to be defined by the user.

$$P = P_{line} \cup P_{back} \; ; \; P_{line} \cap P_{back} = \emptyset \tag{3}$$

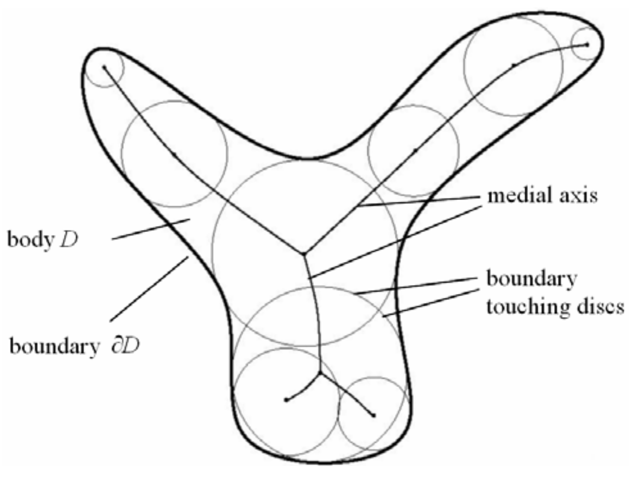

Figure 4: Medial axis of a body in 2D Euklidian space (Wolter 2001)

In Euclidian space the medial axis describes the skeleton of a body (Wolter 2001). All points of the medial axis have the same minimal distance to at least two different points of the boundary ∂D of the body D, as shown in figure 4. The transfer of this medial axis concept to the discrete grid of pixels leads to the burning algorithm (Lindquist and Lee 1996). In analogy to the city block distance in Euclidian space the distance d of two pixels p_{ij} and p_{kl} is defined as the positive difference of the corresponding pixel indices:

$$d(p_{ij}, p_{kl}) = abs(i-k) + abs(j-l) \qquad (4)$$

The burning algorithm defines the minimal distances $d_{\min}$ of all pixels p_{ij} to a background pixel:

$$d_{ij} = \min(d(p_{ij}, p_{kl})) \quad \text{for all} \quad p_{kl} \in P_{back} \qquad (5)$$

For all background pixels these distance values are equal 0. All line pixels have a distance value greater than 0. All line pixels with a distance value 1 are called contour pixels P_{con} and denote the boundary of lines.

$$P_{con} = \{p_{ij} \mid p_{ij} \in P_{line} \wedge d_{ij} = 1\} \qquad (6)$$

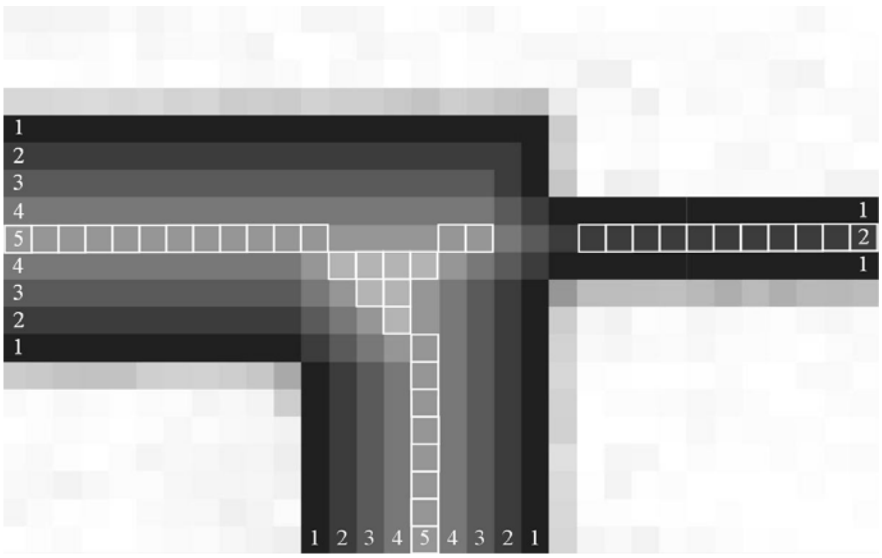

Figure 5: Distance values and medial axis of three lines of a drawing

Figure 5 shows three lines of a drawing detail. The distance values of the corresponding pixel are illustrated by different grey levels. These distance values d_{ij} are used for the determination of the medial axis. The medial axis in the discrete pixel grid is defined as the set of line pixels having no pixel with greater distance value in the direct neighbourhood:

$$P_{maxis} = \{p_{ij} \mid p_{ij} \in P_{line} \wedge d_{ij} \geq d_{i\pm1\,j} \wedge d_{ij} \geq d_{i\,j\pm1}) \qquad (7)$$

The pixels of the medial axis are indicated in figure 5 by a white square. These pixels of the medial axis are used as starting pixels for the line identification process. Starting with these pixels a line is created by adding pixels and rotating the temporary line. This process of adding pixel and rotating the line is illustrated in figure 6 and is performed with the aim of maximizing the sum of distance values of all pixels of the temporary line.

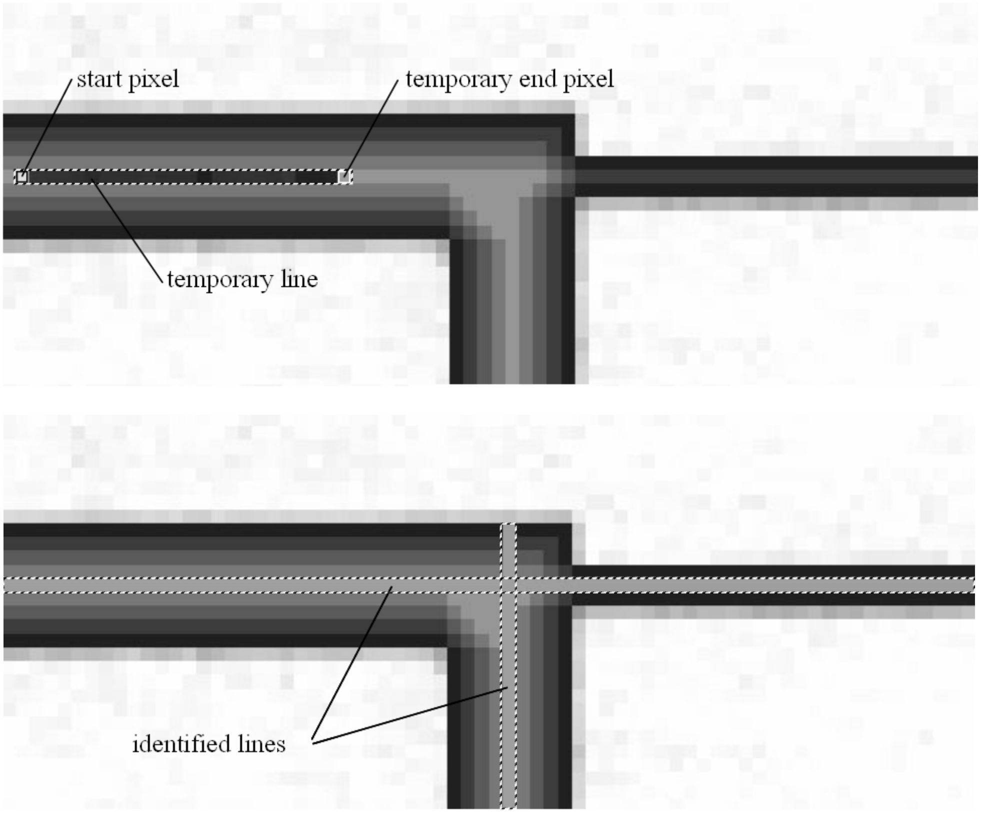

Figure 6: Process of identification of lines in a drawing

This process of adding pixels to the temporary line is aborted, if contour pixels are reached at both ends of the temporary line. If the restrictions of minimum length and minimum distance to other lines are met the temporary line is accepted as an identified line. A curved line in the drawing will be identified as set of lines, which will be transformed by the subsequent topological analysis to a polygon.

3 Geometric and topologic models

The user has to define scale factors and a scale angle by selecting two edges of the drawn building, which have to build a rectangular angle. Based on this input the identified pixel-based lines are converted into lines in the 2D Euclidian space E^2 . This leads to a system of lines called geometric model of a drawing. This geometric model consists of geometric lines and geometric points, as shown in figure 7 for a detail of the drawing.

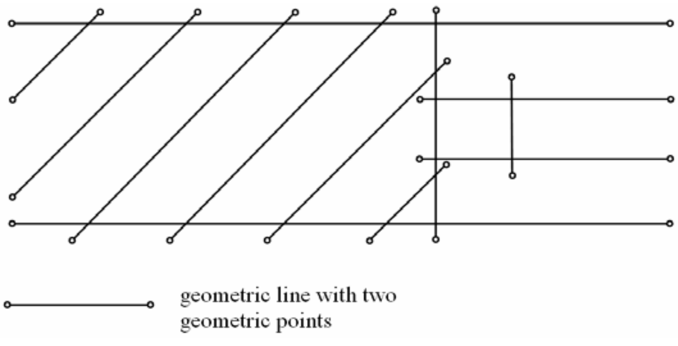

Figure 7: Geometric and topologic model of a drawing detail

A geometric line is defined as the shortest connection of two geometric points. Every line is bounded by exactly two points and every point is connected with exactly one line. The sets of geometric points GP and of geometric lines GL and the corresponding relations GPGL and GLGP between these both sets build a bipartite graph G (Pahl and Damrath 2001):

$$G = (GP, GL; GPGL, GLGP) \text{ mit } GPGL \subseteq GP \times GL; GLGP \subseteq GL \times GP \quad (8)$$

In this geometric model no information about the connectivity of geometric lines and points is yet available. In order to generate this topologic information a topologic model with topologic sections and topologic points is generated. First of all, all intersection points of geometric lines are determined. All geometric and intersection points within a user defined environment of ε are merged to a single topologic point. These topologic points divide geometric lines into topologic sections. Consequently, one topologic point refers to 0 up to n geometric points and one geometric point refers exactly to one topologic point. One topologic section refers to 1 up to n geometric lines and one geometric line refers to 1 up to n topologic lines. These sets of points, sections and lines and the corresponding relations provide all topologic information of the drawing and build a system of four bipartite graphs, shown in figure 8.

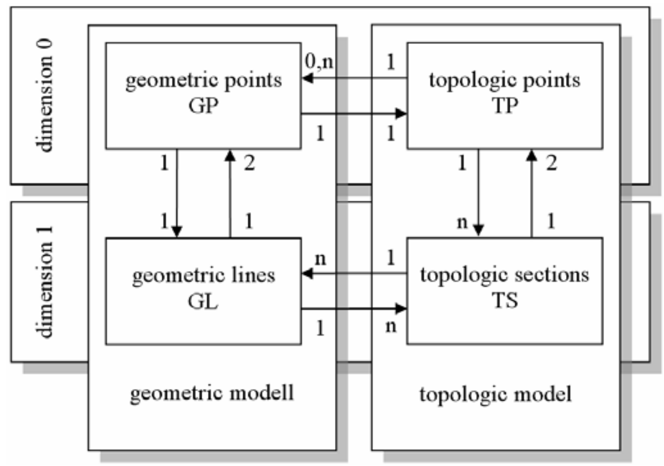

Figure 8: System of four bipartite graphs representing the geometric and topologic model

An example of topologic models is shown in figure 9 for a detail of the drawing. Within a topologic clean-up process all inaccuracies such as small edges or multiple edges are corrected, as shown in figure 10.

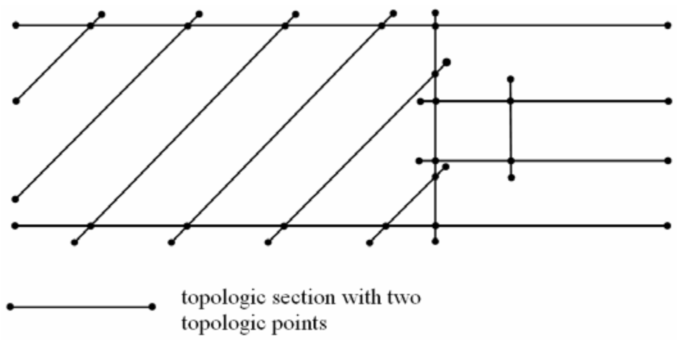

Figure 9: Topologic model of a drawing detail

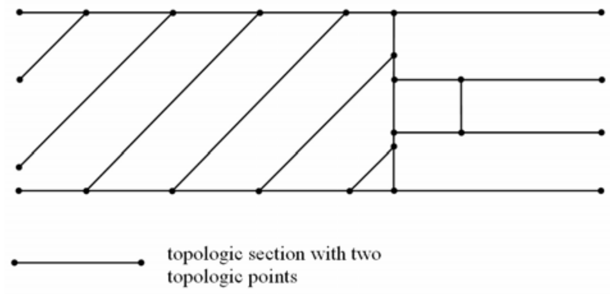

Figure 10: Topologic model of a drawing detail after the topologic clean up process

4 Identification of construction elements

In a ground floor plan each construction part builds a closed loop of topologic sections. This means, no topological end-point, which refers to exactly one geometric point, appears in a closed loop. Consequently, all topological sections with topological end-points are not taken into account for the process of construction part identification and most of the dimension lines and caption texts are eliminated. In the next step hatching lines are identified by their parallelism and missing of 180 degree connections. Closed loops of topological sections with hatching lines inside are identified as wall or column construction parts. Every wall construction part is split into parts with exactly four part points. For the identification of window and door construction parts special topological and geometric criteria are used. In addition, a construction part involves an axis and a thickness. Figure 10 illustrates the identified construction parts for drawing detail.

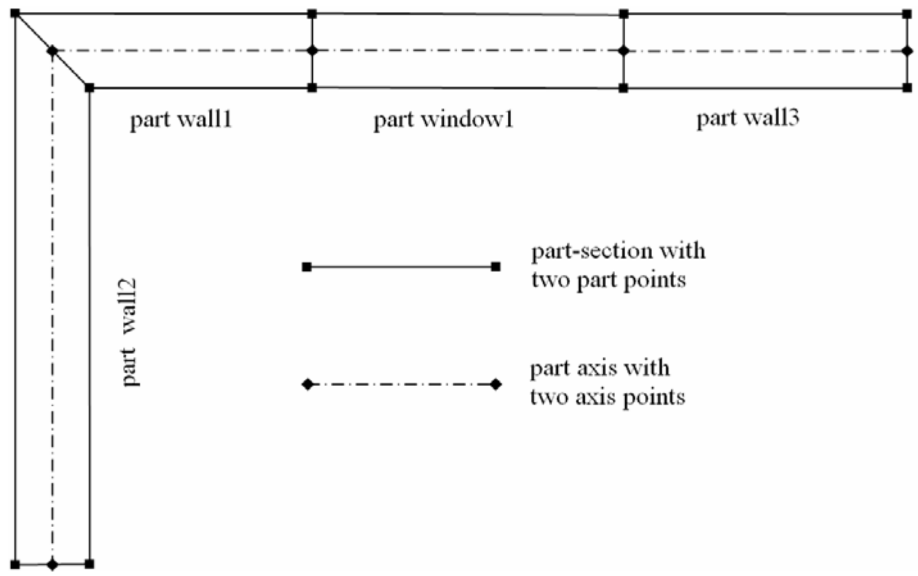

Figure 10: Identified construction parts of a drawing detail

5 Import of construction parts into a product model

For the import of construction parts into a product model independent and dependent parts are defined. Independent construction parts exist without any reference to any other part. In contrast to this, window or door constructions parts are dependent parts, which are defined by a reference to a wall construction part. A ground floor plan implies only the information about the wall parts beside a window or a door part. For the import into a product model the wall parts at both sides of a window or a door part are merged to one independent entire wall part. The corresponding dependent windows and door parts refer to this entire wall part. Figure 11 shows all wall, window and door construction parts as independent and dependent parts.

The identified constructions parts, defined by their axis and thickness, are imported into the geometry kernel of a product model. Additional information about the height between floors and the heights of window parapets or of door lintels has to be defined generally for the whole building. Exceptions of the standard heights have to be specified explicitly for the relevant construction parts.

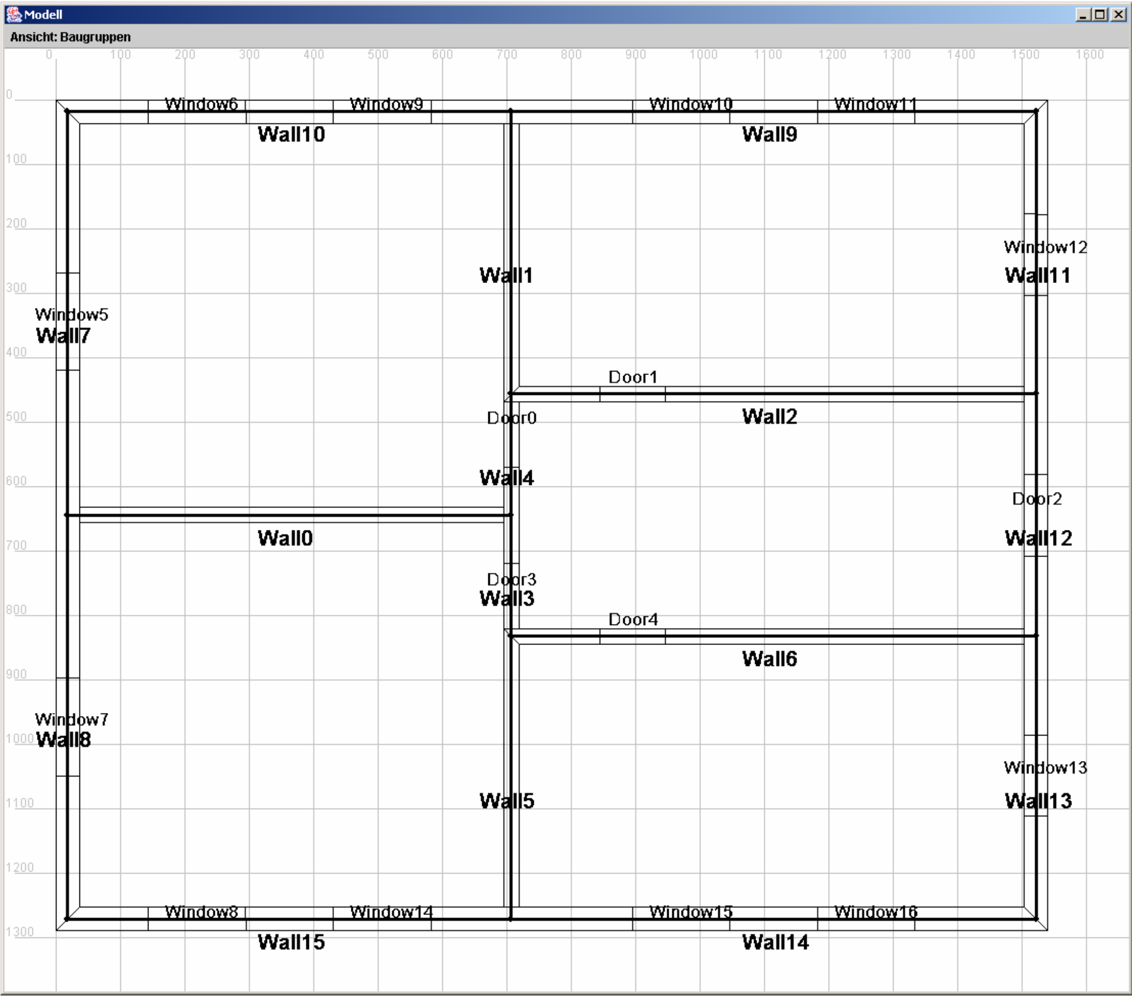

Figure 11: Construction parts for the import into a product model

In addition to geometric and topologic data product information of the building is managed by the product model, which is relevant for all states of design, planning, construction, creation and usage (Neuberg et al. 2002) (Egger and Neuberg 2003). Figure 12 illustrates the 1st floor of the office building as a 3D model imported into the product model.

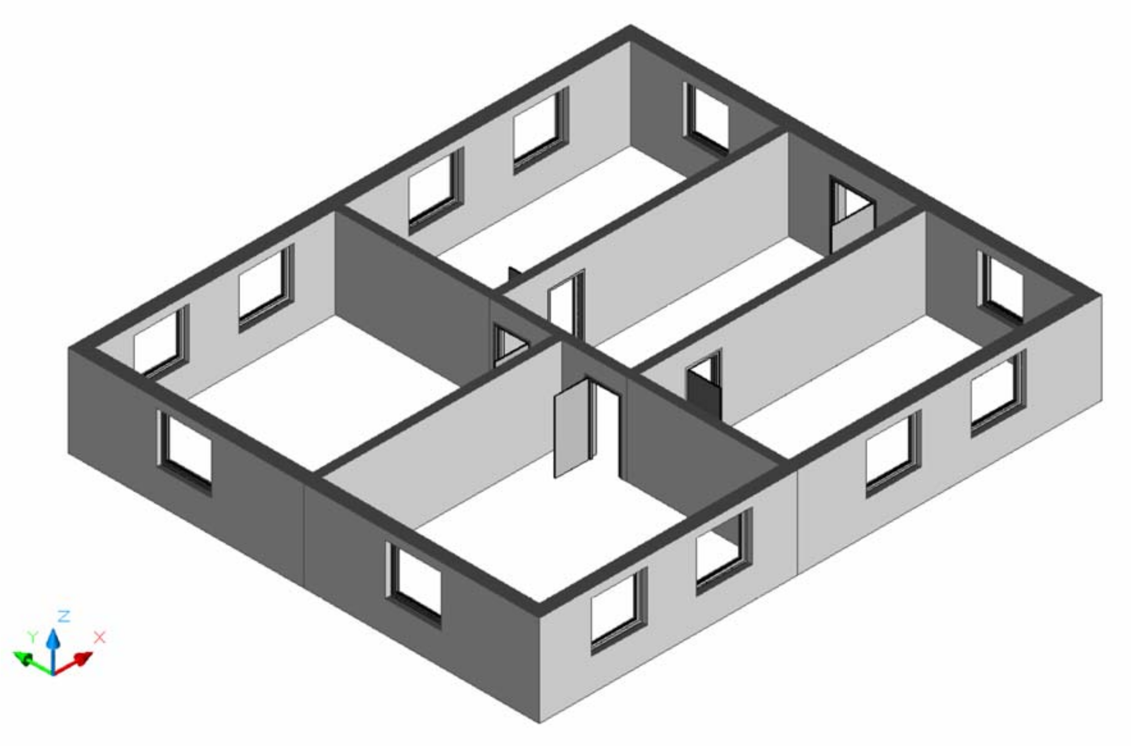

Figure 12: Identified construction parts of the office building imported into the product model

6 Conclusions

In order to facilitate the computer support for the management and reorganisation of existing buildings the gap between paper-based drawings and product models has to be closed. The approach presented in this paper demonstrates the feasibility of identifying construction parts in a ground floor plan and converting these parts into a format suitable for product models. All algorithms are implemented in a prototype software tool, which has to be tested and enhanced for real world problems.

7 References

Berkhahn, V. (to appear 2004). Recalculation of Construction Elements Based on Drawings. Proceedings of "SEMC 2004: The Second International Conference on Structural Engineering, Mechanics and Computation". Cape Town. South Africa.

Berkhahn, V. and Esch, C. (2003). Re-Engineering of Objects in Construction Drawings. In: R. Amor (ed.), "Construction IT Bridging the Distance, CIB Report: Publication 284, Proceedings of CIB W78's 20th International Conference on Information Technology for Construction". Waiheke Island. New Zealand.

Chai, J. and Dori, D. (1992). Orthogonal zig-zag: An efficient method for extracting bars in engineering drawings. In: Arcelli, C.; Cordella, L.P.; Sanniti di Baja, G. (eds.): „Visual Form". New York. Plenum.

DFG (2004). Deutsche Forschungsgemeinschaft: „Vernetzt-Kooperative Planungsprozesse im Konstruktiven Ingenieurbau", DFG_Schwerpunktprogramm 1103, www.dfg-spp1103.de

Dosch, P.; Tombre, K.; Ah-Soon, C. and Masini G. (2000). A complete system for the analysis of architectural drawings. International Journal on Document Analysis and Recognition, Vol. 3, pp. 102-116.

Egger, M. and Neuberg, F. (2003). Ökologische Bewertungsmethoden auf der Grundlage eines Bauwerkmodells“. In: Kaapke, K.; Wulf A (eds.): „Forum Bauinformatik 2003, Junge Wissenschaftler forschen“. Hannover. Germany. Shaker Verlag.

IFC (2003). Industrieallianz für Interoperabilität e.V. (eds.): IFC - Industry Foundation Classes. http://www.iaiev.de/ifc/documents/overview_management.htm.

Lindquist, W.B. and Lee, S.-M. (1996). Medial Axis Analysis of Void Structure in Threedimensional Tomographic Images of Porous Media. Journal of Geophysical Research.

Neuberg, F.; Rank, E.; Ekkerlein, C. and Faulstich, M. (2002). Internet Based Simulation of the Resource Requirement of Buildings. Proceedings of “ECPPM 2002: European Conference of Product and Process Modeling - eWork and eBusiness in AEC". Portoroz. Slovenia.

Pahl, P. J. and Damrath, R. (2001). Mathematical Foundations of Computational Engineering. Springer-Verlag, Berlin / Heidelberg.

Schleinkofer M. (2003). Skelettierungsverfahren zur Extraktion des statischen Systems aus photogrammetrischen Aufnahmen. In: K. Kaapke, A. Wulf (eds.): „Forum Bauinformatik 2003, Junge Wissenschaftler forschen“. Hannover. Germany. Shaker Verlag.

Wolter, F.E. (2001). Cut locus and medial axis in the euclidean space and on surfaces. GDV. University of Hannover. Germany. Laboratory for Engineering Man / Machine Systems. Brown University. Rhode Island. USA. www.lems.brown.edu/vision/Presentations/Wolter

Yoo, J.Y.; Kim, M.-K.; Han, S.Y. and Kwon, Y.-B. (1998). Information extraction from a skewed form document in the presence of crossing characters. In: K. Tombre, & A.K. Chhabra, (eds.): “Graphic Recognition, Algorithms and Systems, Lecture Notes in Computer Science”. Springer-Verlag. Berlin.