Pattern Recognition Letters 35 (2014) 91–100

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

# Automatic analysis and sketch-based retrieval of architectural floor plans

Sheraz Ahmed [a,b,*], Markus Weber [a], Marcus Liwicki [a], Christoph Langenhan [c], Andreas Dengel [a,b], Frank Petzold [c]

[a] German Research Center for Artificial Intelligence (DFKI) GmbH, Trippstadter Straße 122, 67663 Kaiserslautern, Germany
[b] Knowledge-Based Systems Group, Department of Computer Science, University of Kaiserslautern, P.O. Box 3049, 67653 Kaiserslautern, Germany
[c] Chair of Architectural Informatics, Faculty of Architecture Technical University of Munich, Arcisstrasse 21, 80333 Munich, Germany

ARTICLE INFO

*Article history:*
Available online 19 April 2013

*Keywords:*
Floor plan analysis
Structure analysis
Architecture
Wall detection
Symbol spotting
Room detection

ABSTRACT

The contribution of this article is twofold. First, we propose a sketch-based system, namely the a.SCatch system, for querying a floor plan repository. Second, for generating such a repository, a novel complete system for floor plan analysis is presented. The latter system extracts the semantics from existing floor plans. We introduce novel preprocessing methods, e.g., the differentiation between thick, medium, and thin lines and the removal of components outside the convex hull of the outer walls. Especially, the use of the convex hull increases the performance of the final system. The a.SCatch system enables the user to easily access knowledge from past projects. The user searches for semantically similar floor plans just by drawing parts of the new plan. An algorithm extracts the semantic structure sketched by the architect on DFKI's *Touch* & *Write* table. Finally, the extracted structures are compared using the graph matching, and the most similar one is retrieved. In our experiments for floor plan analysis on a reference data set we compare our approach to other approaches available in the literature. We show that our floor plan analysis system outperforms previous systems. Also the performance of the sketch recognition system is quite high. Overall, the performance of the floor plan analysis, as well as of the retrieval, is already good for the use in practice.



## 1. Introduction

During design process, architects use existing and already designed buildings as reference. These reference drawings are used to guide solutions for similar architectural situations. Whenever an architect has to solve a new architectural problem, his first task will be to search for similar projects. By studying one or several previous reference projects the architect tries to derive a solution for his current problem. This is a common approach in architecture, knowing about to use reference projects during the design process and the knowledge of reference projects is an essential skill of an architect.

These days electronic search in architecture is realized by searching for textual annotations. However describing architectural work just using textual information is not sufficient, as verbal descriptions are subjective and often imprecise. Thus a pure textual annotation of floor plans is too fuzzy for an efficient retrieval.

Hence, in order to support an architect in his early design phase, there is a need for a search tool which enables him or her to find similar projects. As already stated, a textual search approach is not sufficient and too fuzzy. Furthermore, as we are dealing with visual information it is more intuitive to formalize a query employing a visual query language. The language should be an abstraction of an original floor plan symbolism, in order to have an intuitive way for architects to formalize a search query.

Langenhan proposes a semantic structure to describe the content of a floor plan based on functional and spatial relations of different structural entities. This formalization results in a graph representation which can be used for the retrieval process. The a.SCatch system enables the user to easily access knowledge from past projects. The user searches for semantically similar floor plans just by drawing parts of the new plan.

However, it is a well known problem that it is difficult to generate the semantic database for already existing reference paper floor plans, also known as the bootstrapping problem. Therefore, our second major contribution is an automatic floor plan recognition system which analyzes the floor plans and finally retrieves the corresponding semantic information. Usually, floor plan analysis systems consist of information segmentation, followed by

* Corresponding author at: German Research Center for Artificial Intelligence (DFKI) GmbH, Trippstadter Straße 122, 67663 Kaiserslautern, Germany.
*E-mail address:* sheraz.ahmed@dfki.de (S. Ahmed).

0167-8655/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.patrec.2013.04.005

structural analysis and finally, semantic information extraction and alignment. The retrieved structural and semantic information can be saved in a repository for later access during retrieval.

Note that this article is an extended version of the conference contribution by Weber et al. (2010), where the a.SCatch system was initially proposed. While Weber et al. (2010) provided only a short description of the retrieval system, this article presents more details on all involved processes. Furthermore, the floor plan recognition methods introduced by Ahmed et al. (2011a) are elaborated and evaluated in this paper as well. Finally, we propose novel post-processing techniques for the semantic floor plan analysis and report on results of the floor plan recognition as well as sketch recognition and floor plan retrieval.

The novel key techniques of our approach are: (i) the segmentation into thick, *medium*, and thin walls to improve preprocessing techniques; (ii) the extraction of the building boundary removing many false positives during text extraction; (iii) detection of wall edges using convex-concave hypothesis; (iv) use of patch based approach coupled with Speeded Up Robust Features (SURF) for finding doors locations in the given floor plan.

The rest of this paper is organized as follows. First, Section 2 gives an overview of the related work. Subsequently, Section 3 introduces the concepts of the a.SCatch system and discusses the details of the floor plan analysis system. Section 4 shows the performed experiments. Finally, Section 5 concludes the work.

## 2. Related work

This section provides an overview of related work to the different techniques used in this paper. Related work can be categorized into architectural background, sketch-based interfaces, symbol spotting, complete floor plan analysis systems, and graph matching.

### 2.1. Architectural background

Since the middle of the 1990s the approach of applying Case-Based Reasoning (CBR) to design and architectural tasks has been known as Case-Based Design (CBD). The case-base contains information on buildings that have already been built or designed, enabling the computer to adapt solutions accordingly, on its own or with help from the architects. Table 1 provides a brief overview of some CBD systems based on two studies published by Heylighen and Neuckermans (2001) and by Richter et al. (2007) regarding the proposed approach. The marked fields show whether the appropriate feature was realized in the concept.

The study by Richter et al. (2007) identifies an acquisition bottleneck in putting complete case descriptions (problem and solution) into the case-base. We assume this is due to a lack of adequate input strategies, indexing methods and knowledge management procedures. First of all, a user interface should support the graphical sketch-based workflow of architects combined with textual, schematic and tabular input strategies. Secondly, a lightweight indexing strategy is needed in contrast to the overall data storage method used. Thirdly, the problem and solution descriptions need to be stored according to the CBR paradigm. Most of the CBD prototypes do not properly implement this fundamental CBR attribute.

### 2.2. Sketch-based interfaces

Sketches are widely used in engineering and architectural fields as they are a familiar, efficient and natural way of expressing certain kinds of ideas. Feng et al. (2008), proposed an 2D dynamic programming approach for analyzing hand-drawn electronic circuits. Sezgin et al. (2001) introduced a system that combines multiple sources of knowledge to provide robust early processing for freehand sketching.

Sim-U-Sketch is a sketch-based interface for Simulink[1] (Kara and Stahovich, 2004) where users can construct functional Simulink models simply by drawing sketches on a computer screen. To support iterative design, Sim-U-Sketch allows users to interact with their sketches in real time to modify existing objects and add new ones.

The *COMIC* system (Os and Boves, 2003) is a large European project that studies multi-modal interactions in design applications using pen and speech. In multi-modal system methods, such as mode detection by Willems et al. (2005), are sufficient to improve the usability of these systems.

Spatial-Query-by-Sketch proposed by Egenhofer (1996) describes a visual spatial query language for geometric information systems. Yaner and Goel (2006) examines the retrieval and mapping tasks of visual analogy as constraint satisfaction problems.

### 2.3. Symbol spotting

A main issue in the floor plan analysis is symbol spotting. Therefore, we list some related work in symbol spotting in this section.

In the past, different pattern recognition techniques have been applied to symbol spotting. Belkasim et al. (1991), Li and Shen (1991), Adam et al. (2000) and Arajo and Kim (2007) used feature based description for the purpose of symbol spotting. Similarly symbol spotting based on structural representation of documents has been used by Lladoós et al. (2001) and Yan and Wenyin (2003). Furthermore, Tabbone et al. (2003) performed symbol spotting based on image segmentation. However, segmentation itself leads to errors, which are then propagated to the recognition.

Another idea to address symbol spotting is to use a vectorial image to spot the symbol rather than using a raster image. Messmer and Bunke (1996) and Rusiñol and Lladós (2006) used vectorized image for symbol spotting. In addition to vectorization, Rusiñol et al. (2010b)used indexing techniques to increase the efficiency of symbol spotting techniques. To further increase the scalability of the method Rusiñol et al. (2010a) used relational indexing of vectorial primitives. Dutta et al. (2011) proposed a method using hashing the shape descriptors of graph paths.

Nayef and Breuel (2010) used geometric primitives as feature points. These feature points are then searched using geometric matching algorithm. To further increase the performance as well as the accuracy of the method, Nayef and Breuel (2011) used statistical grouping for segmenting parts from line drawings. After grouping, symbol spotting is performed using the method by Nayef and Breuel (2010).

### 2.4. Floor plan analysis

In past, floor plan analysis has been performed for different purposes. Aoki et al. (1996) and Lladós et al. (1997) analyzed hand sketched floor plan and generated respective CAD representation. Whereas, Dosch and Masini (1999), Dosch et al. (2000), Lu et al. (2007) and Or et al. (2005) focused on analysis of 2D diagrams of floor plans so that their respective 3D model can be regenerated.

Wessel et al. (2008) proposed a method for extracting the room connectivity graphs from 3D architectural models. Based on this

[1] Simulink is an environment for multidomain simulation and Model-Based Design for dynamic and embedded systems. http://www.mathworks.com/products/simulink/: Last accessed 04/02/2010