General Diagram-Recognition Methodologies

Dorothea Blostein
Computing and Information Science
Queen’s University, Kingston Ontario, Canada, K7L 3N6
blostein@ucis.queensu.ca

The field of diagram recognition faces many challenges, including the great diversity in diagrammatic notations, and the presence of noise and ambiguity during the recognition process. To help address these problems, research is needed into methods for acquiring, representing, and exploiting notational conventions. We review several frameworks for diagram recognition: blackboard systems, schema-based systems, syntactic methods, and graph rewriting. Next we discuss the need for a computationally-relevant characterization of diagrammatic notations, the need to exploit soft constraints during diagram recognition, and the possibility that diagram generators may provide a useful source of information about notational conventions.

Keywords: diagram recognition, notational conventions, diagrammatic notations, contextual information, diagram classification, blackboard systems, schema-based systems, graph rewriting

1. Introduction

A variety of diagrammatic notations are widely used in society; examples include notations for music, math, architecture, and logic circuits. These diagrammatic languages use two-dimensional arrangements of symbols to transmit information. Understanding a diagram involves two activities: symbol recognition and symbol-arrangement analysis. Here we focus on the latter: analysis of the spatial arrangement of a group of symbols, relative to the given notational conventions of a 2D language, to recover the information content of a diagram. Important open questions in this area include:

- How should notational conventions be represented? We need systematic methods for representing conventions in a format suitable for computation. Current diagram recognition systems tend to use ad-hoc methods for representing notational conventions (e.g. [BlBa92] reviews music-notation understanding). Comprehensive expositions of notational conventions have been published (e.g. [Read79] for music notation); these present information in a style suited for human readers, but unsuited for computation. Where computationally-oriented descriptions of notations have been developed, most are oriented toward generation of the notation rather than recognition of the notation (e.g. [Rous88], [Knut79]).

- How can noise and uncertainty be handled? How should context be applied to reduce errors and uncertainty in segmentation and symbol-recognition?

- How can the great diversity of diagram types be handled? We need general, systematic methodologies for organizing recognition systems, representing
notational conventions, and using contextual information to reduce errors. Ideally, such research would eventually result in a recognition system that is reconfigurable for various diagrammatic notations.

Much of the current research in diagram recognition is directed at recognizing some particular type of diagram. A comparison among recognition systems (Section 4) reveals common themes in the strategies used to design and implement recognition systems. We hope that over time this collective experience will give rise to a technology for general diagram recognition. An appealing analogy is provided by compiler technology, where general techniques for parsing and code generation greatly simplify the task of constructing compilers for new source and target languages. Diagram-recognition methods are difficult to generalize, due to the great diversity among diagrammatic notations, and due to the tremendous problems arising from noise and uncertainty in the input. Nevertheless, our research community as a whole would benefit from increased sharing of design and implementation strategies. Currently there are no criteria for choosing an appropriate recognition framework, or for determining how best to represent and apply notational conventions.

2. Six Diagram-Recognition Processes

Diagram recognition consists roughly of the following processes:

1. early processing -- noise reduction, de-skewing, etc
2. segmentation, to isolate symbols
3. recognition of symbols
4. identification of spatial relationships among symbols
5. identification of logical relationships among symbols
6. construction of meaning

All of these processes require knowledge of notational conventions. For example, noise reduction methods (1) must preserve the presence of small or thin symbols, like decimal points in math and duration dots in music. Segmentation (2) needs information about how symbols overlap; an example is the separation of notes from staff-lines in music. Symbol recognition (3) needs information about symbol appearance; perhaps a font defining fixed symbols, and structural descriptions for parameterized symbols. The identification of spatial relationships (4) requires information about which spatial relationships are significant for encoding information. The identification of logical relationships among symbols (5), and construction of meaning (6), rely heavily on knowledge of notational conventions. Further research is required to clarify how notational conventions should be represented and used.

World knowledge plays an important role in diagram recognition. For example, knowledge of disassembly and kinematics is used for the recognition of engineering drawings [VaTo94]. Similarly, human recognition of music notation uses the reader's knowledge of music theory and compositional styles. The acquisition, representation, and use of world knowledge is a broad and interesting topic, but one that lies beyond the scope of this paper.

While we find it useful to discuss diagram recognition in terms of the above six processes, the processes are not necessarily clearly delineated in an implementation, and