Visual Structure Analysis of Flow Charts in Patent Images

Roland Mörzinger, René Schuster, András Horti, and Georg Thallinger

JOANNEUM RESEARCH Forschungsgesellschaft mbH
DIGITAL - Institute for Information and Communication Technologies
Steyrergasse 17, 8010 Graz, Austria
<firstname.lastname>@joanneum.at

Abstract. This report presents the work carried out for the flow chart recognition task in the course of the CLEF-IP 2012 competition. The goal is to obtain structural information of flow charts based on the visual content of the images. To this end, for each flow chart a list of its nodes and their interconnections, i.e. its edges, is extracted and the type of the nodes and edges and attached text is recognized. The automatic recognition task is done in three stages: (1) flow chart image pre-processing using connected component analysis, morphological filters and line segmentation, (2) identification of nodes, junction points, end points and edges and (3) recognition of text, geometric node types and edge directions.
Examples demonstrate good recognition results obtained for 100 tested flow chart images.

Keywords: patent, flow charts, images, technical drawings, structure analysis

1 Introduction

In traditional engineering drawings and diagrams, algorithms, operations and processes are frequently represented as flow charts. In patents, these drawings generally are accessible only in image format. For automatic querying the huge information content of the flow charts available in patents, it is important to convert the information in the images into a high-level description [5]. This problem usually involves techniques in the field of image binarization, segmentation, shape extraction and recognition of text and geometric components [4].

This paper describes our approach for automatically analyzing the visual structure of flow charts and our participation in the flow chart recognition task [1] in the course of the CLEF-IP 2012 competition.

At this, it is assumed that a flowchart can be interpreted as a graph with a set of nodes and edges [3]. To semantically process the data therein, the extracted information should contain:

1. the number of nodes,

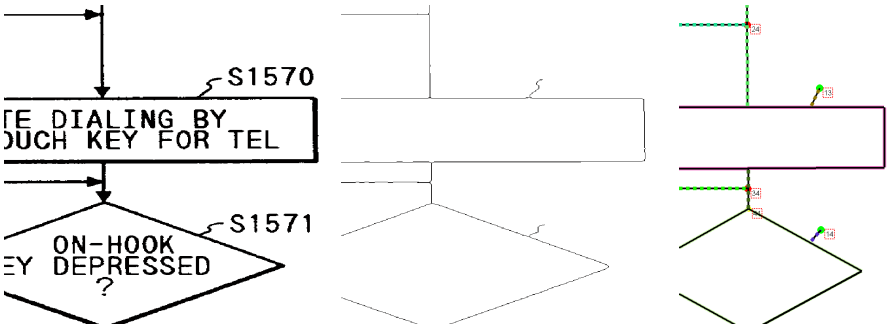
2. the type of each node (e.g. rectangle, diamond, oval, etc),
3. the text annotations (if any) within each node for creating the link between the image and the patent text,
4. the interconnections, i.e. edges, between the nodes and
5. the type of the edges (e.g. continuous, dotted, etc.).

The rest of this paper is organized as follows: Section 2 describes the applied image processing methods for producing the above mentioned metadata and results are presented in Section 3.

2 Visual Structure Analysis

A summary of the flow chart recognition process is shown in Figure 2, understandably a flow chart itself. First, textual descriptions in the input image are extracted using optical character recognition, followed by pre-processing, line segmentation and grouping. Next, junctions and end points are detected and based on that, nodes and their interconnections (edges) are recognized. Finally, the type of the nodes and the direction of the edges is computed and the visual information content from the flow chart is provided in a textual graph representation (for the format see [1]).

The quality of Figure 2 was deliberately modified in a way to match the look&feel and (sometimes bad) image quality of many technical drawings found in patents. When viewed in full detail, lines are frequently frayed, non-contiguous and not strictly vertical or horizontal, which clearly presents challenges for image processing. The following subsections explain the processing steps of our approach necessary for recognizing the visual structure of flow charts.



(a) (b) (c)

Fig. 1. Example showing details of different processing steps. The black and white input image (a) is pre-processed and based on the resulting cleaned and thinned image (b) the final visual structure (c) is recognized. It shows nodes (solid colored border), edges (dashed line), junction points (red filled circle), end points (green filled circle) and the nodes' IDs (numbers in red boxes). Best viewed in color

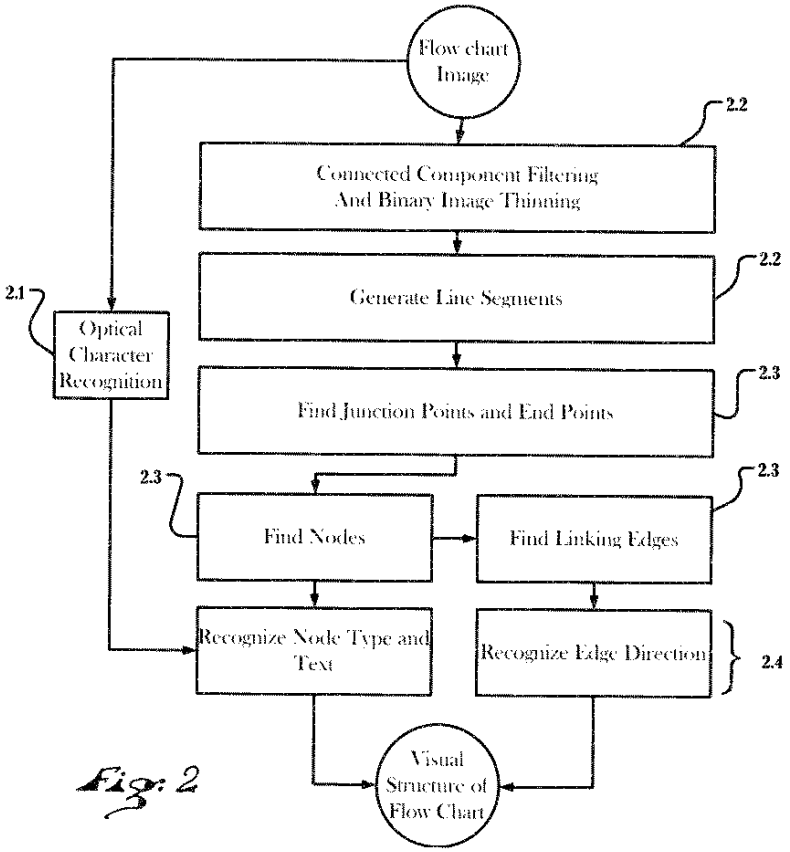


Fig. 2. Overall flow chart recognition procedure. Best viewed large for detail.

2.1 Text, Label and Reference Detection

A commercial optical character recognition (OCR) software [2] is used to extract text from the flow chart images. String matching using a regular expression is performed for identifying possible references, like "Fig. x". The locations (image coordinates) of the extracted text blocks are kept for subsequent association with detected nodes and edges.

2.2 From Binary Image to Line Segments

First, in the binary input image all connected components with a small number of pixels and aspect ratio typical to characters (as opposed to lines) are removed. Second, the image that is now cleaned from text-like fragments is subjected to a morphological close and binary image thinning operation, see Figure 1(b). Third, edge points are identified and linked to segments. By respecting a specified

minimum line length and maximum deviation from the original data, the pixel image can now represented as a list of linked line segments.

2.3 Detection of Junctions, End Points, Nodes and Connecting Edges

Junctions and end points can be easily found by scanning the linked line segments from the previous step for areas where three or more segments meet (junction) or where a segment has no further connection (end point). In many cases, end points are actually the end of wiggly edges that connect the nodes of the flowchart with their labels.

Next, the nodes of a flow chart are detected by iteratively finding for each segment another linking segment that is close enough (allowing for nodes with small gaps between segments) and that has the smallest angle between them. A combination of linked segments that meets certain criteria, such as a maximum number of segments per node and sum of angle values, constitutes a node.

Subsequently, connecting edges are derived from the remaining segments that that link nodes, junctions or end points. These edges may obviously consist of multiple segments.

2.4 Recognition of Node Type and Edge Direction

The goal is to classify 10 types of nodes (oval, rectangle, double-rectangle, parallelogram, diamond, circle, point, cylinder, no-box and unknown). Their exact definition is given in the task description document [1]. The node types "no-box" and "point" are a direct result of the methods described in Section 2.3. For the classification of the remaining node types, the input is a sequence of segments with start and end point coordinates. Based on those segments, the following features are calculated:

- Number of edges
- Ratio between maximum and median edge length
- Ratio between minimum and median edge length
- Ratio between maximum and minimum angle
- Median of the angles
- Sum of the angles
- TopPositionRatio: normalized ratio between top most and second top most point
- Normalized ratio between right most and second right most point
- Extent: ratio between the area of the bounding box and the convex hull

By using 400 annotated examples, discriminating features and their statistics have been empirically determined for each type. The statistics consist of average, standard deviation, minimum and maximum values for each feature and node type. Figure 3 plots 3 features over 3 node types. The characteristics of the data show that it is possible to classify the node types. With help of these statistics a

score for each class is calculated and the maximum score results in the classified node type. If the maximum score is below 50% the node type is declared as "unknown". An unknown node type is an indication of a possibly inaccurate preceding node detection result and as a consequence the node can be discarded (c.f. runs with node type filter in Section 3).

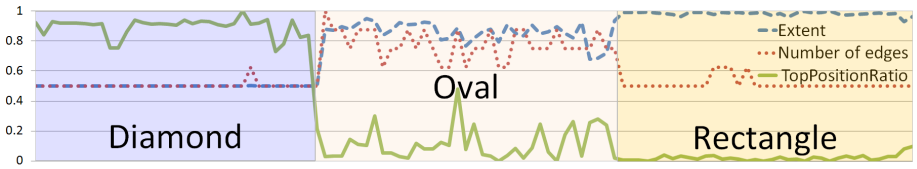


Fig. 3. Three features over 106 examples with three different node types.

The edge direction is estimated by comparing the number of black pixels of the edge segments. A window is centered at the end of the edges and the edge is directed if one the windows has clearly more pixels than the other. The related thresholds have been determined using annotations (c.f. Section 3).

3 Results and Evaluation

For the flow chart recognition task of the CLEF-IP 2012 competition [1], 50 images containing flowcharts and the corresponding annotations were provided. The images were used for developing the recognition system and the annotations, i.e. the number and type of the nodes and edges, were used to tune data-specific parameter values.

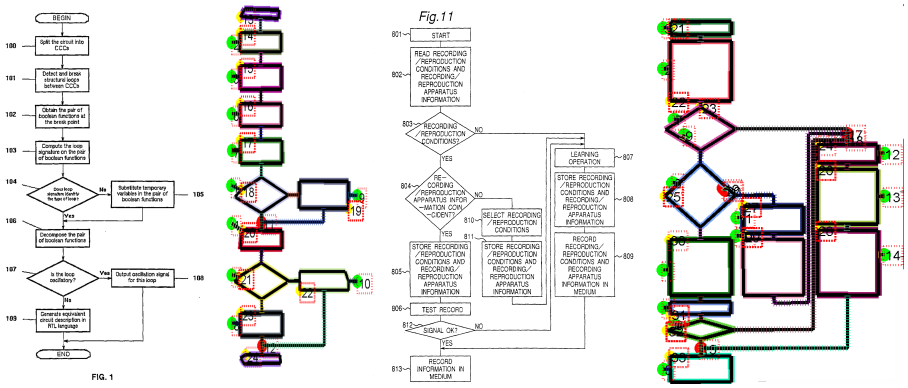
For evaluating our approach and different configurations thereof, we produced a set of runs described in Table 1.

ID	RUN	Description
1	joanneum_flow_tomato	default settings with node type filter [default]
2	joanneum_flow_zuchini	tomato without node type filter
3	joanneum_flow_carrot	tomato with adapted edge direction detection
4	joanneum_flow_romanesco	tomato with adapted junction detection
5	joanneum_flow_basil	tomato with adapted end point detection
6	joanneum_flow_radish	tomato with adapted node detection
7	joanneum_flow_pepper	tomato with adapted line segmentation 1
8	joanneum_flow_rucola	tomato with adapted line segmentation 2

Table 1. Different runs submitted for the flow chart recognition challenge. Details on the mentioned filter and detection methods are given in the Section 2.

All of the different configurations of our flow chart recognition system were applied on a test set of 100 images. Details on the evaluation and the applied *most common subgraph* metric can be found in the task description document [1]. At the time of writing this paper, quantitative evaluation results for the flow chart recognition track in CLEF-IP 2012 were not available yet.

Nevertheless, the examples shown in Figure 4 and 5 should give an impression of the quality of the flow chart recognition system.



(a) (b) (c) (d)

Fig. 4. Two examples with parts of the flow chart recognition results. For the input (a, c) flow charts, the output (b, d) of the detection methods for nodes (solid colored border), edges (dashed line), junction points (red filled circle) and end points (green filled circle) is shown. The numbers in red boxes are IDs of detected nodes. The type of the nodes, the direction of the edges and any attached text is also recognized, but not presented in these output images.

4 Conclusions

This paper presented the experiments for our participation in the CLEF-IP 2012 flow chart recognition challenge [1]. Structural information of flow charts, such as nodes, their interconnections and annotating labels were obtained based on image processing technology using the visual image content only. Although the nodes and interconnections of flow charts seem to be easily recognizable by humans, automatic processing is quite a challenge. On the pixel level, the black lines are frequently non-contiguous, frayed, of varying width and not strictly vertical or horizontal. Flow chart images show different types of nodes (circles, boxes, parallelograms, etc), edges (directed, undirected), typewritten and sometimes handwritten labels. When text is not clearly separated from nodes or lines, difficulties increase. Generally, finding proper values for critical thresholds, such as the minimum length of segments and parameters for morphological filtering,

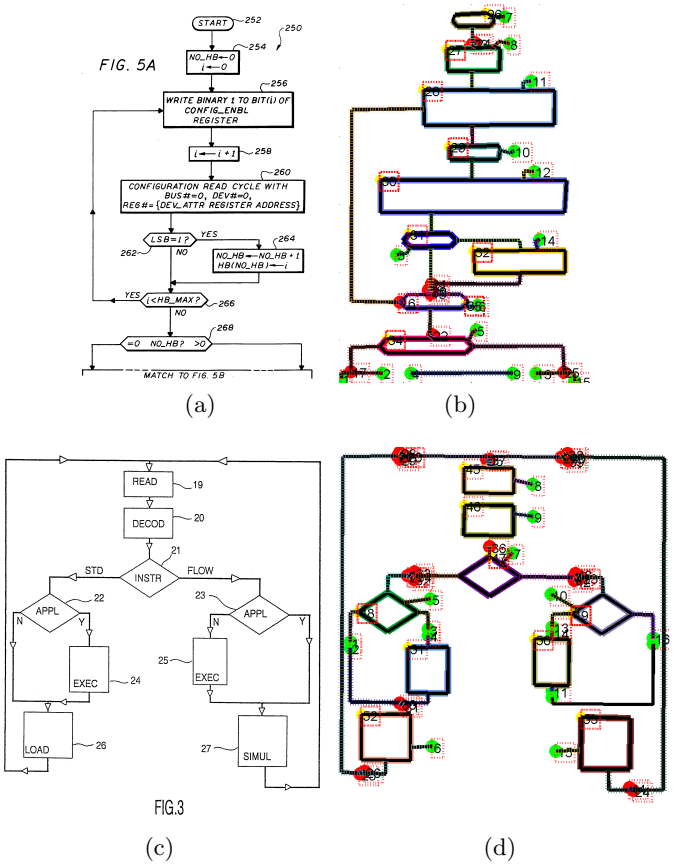


Fig. 5. Two exemplary results of flow chart recognition on challenging data. The dashed lines in the lower part of the input flow chart (a) and the variety of directional arrows (a, c) lead to an inaccurate segmentation of lines and edges (b, d). See Figure 4 for detailed explanation of the result images.

is one of the most crucial parts. For that purpose, the provided annotations have proved very useful. Examples demonstrate good recognition results obtained for 100 tested flow chart images.

Acknowledgments

This work was supported by the Austrian Research Promotion Agency (FFG) FIT-IT project IMPEx [1] Image Mining for Patent EXploration (No. 825846).

[1] http://www.joanneum.at/?id=3922

References

1. CLEF-IP flow chart recognition task 2012. Available online at `http://www.ifs.tuwien.ac.at/~clef-ip/flowcharts.shtml`, visited on Aug. 2012.
2. Transym OCR engine. Available online at `http://www.transym.com/`, visited on Aug. 2012.
3. M. Lupu, R. Mörzinger, T. Schleser, R. Schuster, F. Piroi, and A. Hanbury. Patent images - a glass encased tool / opening the case. In *Proc. of iKnow Conference*, 2012.
4. B. G. Vasudevan, S. Dhanapanichkul, and R. Balakrishnan. Flowchart knowledge extraction on image processing. In *IJCNN*, pages 4075–4082. IEEE, 2008.
5. Y. Yu, A. Samal, and S. C. Seth. A system for recognizing a large class of engineering drawings. *IEEE Trans. Pattern Anal. Mach. Intell*, 1997.