# Chapter 13
# Flowchart Recognition in Patent Information Retrieval

**Marçal Rusiñol and Josep Lladós**

**Abstract** In this chapter, we will analyse the current technologies available that deal with graphical information in patent retrieval applications and, in particular, with the problem of recognising and understanding information carried by flowcharts. We will review some of the state-of-the-art techniques that have arisen from the graphics recognition community and their application in the intellectual property domain. We will present an overview of the different steps that compound a flowchart recognition system, looking also at the achievements and remaining challenges in such a domain.

## 13.1 Introduction

A patent can be defined as a legal title protecting a technical invention for a limited period. Patent documents consist of three parts mainly [10]. First, a front page presenting general information about the patent, such as the title, the summary of the invention, the name of the inventors, etc. Second, the technical description, which details the technical problem the invention solves as well as the state of the art and the novelty of the invention. Finally, a claims section that defines the intellectual property (IP) protection rights, i.e. a clear description of what is legally protected. In each of these parts, drawings can be (and are often) used to provide an accurate detailed description of intermediate parts of the invention.

Since patent documents include both technical and legal information, conducting a patent search is of extreme importance for several purposes [11]. The technical part of patents, as in the case of scientific publications, defines the state of the art for a given problem and can be used to find out what already exists and to check the novelty of a given invention. Concerning the legal aspects, they can also be used in order to assess the freedom to operate, i.e. make sure we are not infringing someone else's IP rights, or to check whether someone might be infringing our own IP rights.

M. Rusiñol (✉) • J. Lladós
Dept. Ciències de la Computació, Computer Vision Center, Edifici O, UAB, 08193 Bellaterra, Spain
e-mail: marcal@cvc.uab.cat; josep@cvc.uab.cat

However, performing searches in the patent's content might not be a straightforward task.

First of all, there is the scale factor. Just in 2013, 265,900 European patent filings were made at the European Patent Office (EPO),[1] representing a 2.8 % growth with respect to the last year. And as of today, nearly 80 million patent documents worldwide are available through the publicly available patent database Espacenet.[2] Conducting efficient and effective searches in such large-scale and ever-growing scenarios is by itself a difficult problem.

Secondly, we have the semantic problem. Patents being written in an 'unstructured' manner in natural language entail the same problematics of any textual search from the information retrieval (IR) field. When searching for a patent, the user has to select a list of keywords that define the invention he/she is looking for. Finding keyword synonyms, avoiding the use of homonyms and using Boolean operators to regroup the terms in order to cast a good query are of critical importance [10].

Finally, it is worth to mention that not all the information in a patent document is conveyed by textual elements. Drawings in patents play an important role since in many patent filings the technical details are depicted rather than being explicitly written in textual format. Drawings can be of different nature, including line drawings, figures, diagrams, flowcharts, plots, etc. As pointed by many authors, e.g. Bhatti and Hanbury in [2], Hanbury et al. in [15] and Lupu et al. in [26], the inspection of visual information conveyed by such drawings is becoming overwhelmingly important in order to assess the novelty of a submitted patent. However, nowadays most of the patent search applications fail to exploit non-textual information [1, 23].

It is worth to note the efforts made within the CLEF initiative concerning the problem of dealing with non-textual information in the IP domain. Until 2011, the CLEF-IP track served as a benchmarking activity on prior art retrieval focusing only on textual patent documents. However, in 2011 two image-based tasks were added [30]: one devoted to find patent documents relevant to a given patent document which contained images and another aimed at categorising patent images into predefined categories of images (such as graphs, flowcharts, drawings, etc.). In CLEF-IP 2012 [31], a new image-based task was proposed: the flowchart recognition task dealing with the interpretation of flowchart line drawing images. The participants were asked to extract as much structural information as possible from these images and return it in a predefined textual format for further processing for the purpose of patent search. Three different institutions participated in such a task [29, 34, 42]. We will overview such approaches and put them in context throughout the rest of this chapter. We will analyse the current technologies available that deal with graphical information in the patent retrieval application. We will review some of the state-of-the-art techniques arisen from the graphics recognition community and their application in the IP domain. Specifically, we will focus on the problem of recognising and understanding information carried by flowcharts.

---

[1]http://www.epo.org/about-us/annual-reports-statistics.html.

[2]http://www.epo.org/searching/free/espacenet.html.

The rest of the chapter is organised as follows. In Sect. 13.2 we will overview the state of the art within the graphics recognition field and its applications to the IP domain. In Sect. 13.3 we will present an overview of the different steps that compound a flowchart recognition system. Finally, in Sects. 13.4 and 13.5 we will present the remaining challenges and our concluding remarks, respectively.

## 13.2   Graphics Recognition: From Visual Similarity to Interpretation

The use of non-textual information is pervasive in the whole IP domain. We can see in Fig. 13.1 some examples of graphical entities found in IP documents such as trademark registrations, design applications or drawings in patents such as flowcharts, molecules, mechanical drawings or electronic diagrams. The management of such graphical entities in the prior art search is of crucial importance since they depict either directly what the owner wants to protect, as in the case of trademarks or designs, or essential details of the invention under patent protection.

The graphics recognition community has been proposing methods for describing and recognising symbolic information for many years [18], and many symbol recognition proposals have a direct application in the IP domain. We can cite, for example, some works dealing with trademark description [19, 32], molecule recognition [13, 36], symbol recognition in mechanical drawings [6] or electronic diagrams [52]. However, most of the symbol recognition techniques present an important drawback. The symbols to recognise have to be isolated so that the descriptors represent them globally. Globally describing graphical information might be already suitable for recognising trademarks of molecule depictions, but is not usually desirable when dealing with graphical entities formed by a compound of symbols and their relationships among them [33], as in the case of electronic diagrams, mechanical drawings or flowcharts.

However, in general, most of the published approaches dealing with patent image retrieval globally describe graphical information and then follow either a content-based image retrieval (CBIR) paradigm [21] or a classification paradigm [8, 28]. In such paradigms the retrieval or recognition is thus performed in terms of a similarity measure between the query image and the images in the corpus. For example, in [17], patent drawings are represented using attributed graphs and the retrieval task is then casted as a graph similarity computation. Methods like [7, 46, 47] and [40] base the image description on histograms encoding the centroid positions at different levels. Such descriptors can be understood as a special case of a quad-tree [37] encoding of the image under analysis. The retrieval part just relies on the Euclidean distance between the query and corpus descriptors. In [43], the PATSEEK framework is presented in which patent drawings are described by means of edge orientation autocorrelograms [27].
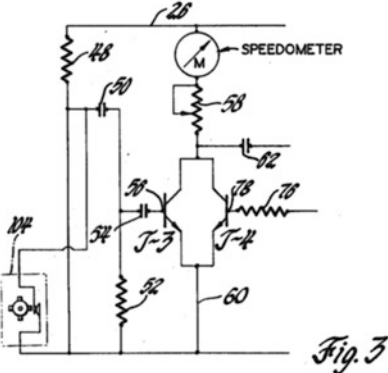
Trademarks                                    Designs

Fig. 5
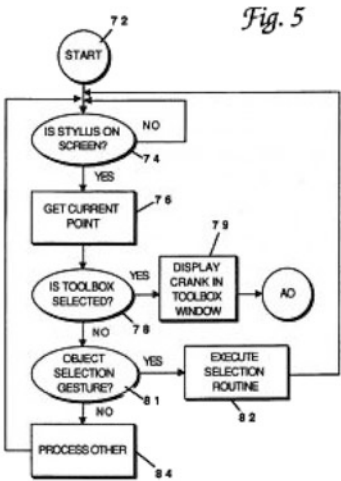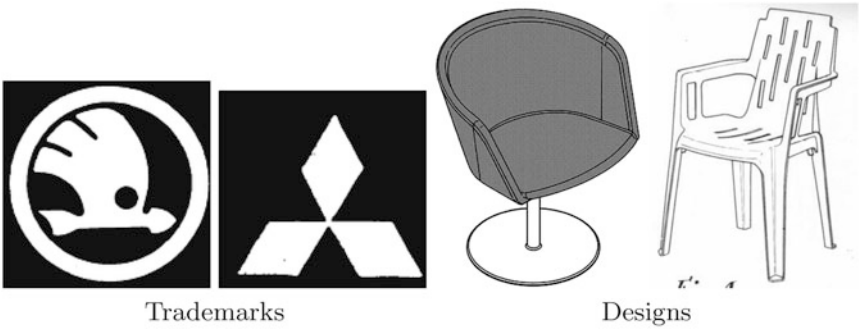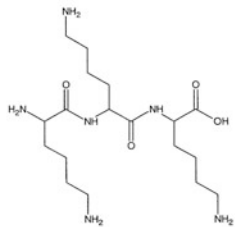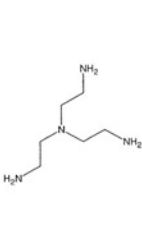
Flowchart                                     Molecule
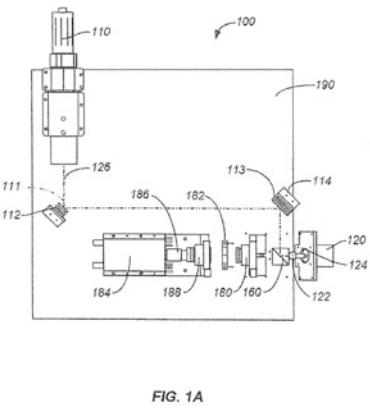
Mechanical drawing                            Electronic diagram

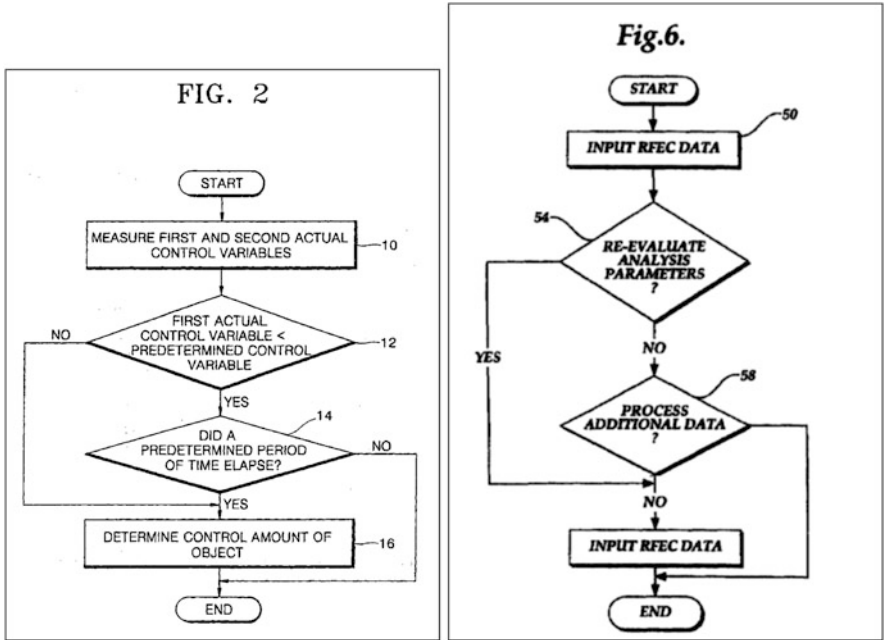**Fig. 13.1** Examples of drawings in the IP domain

**Fig. 13.2** Example of two visually similar flowcharts

Nonetheless, the CBIR paradigm might not be the most suitable tool to provide an image search in the intellectual property domain. In order to assess whether an invention is new or has already been submitted, the patent professional should look for images that depict the *same concept* [47] instead of images that *look visually similar* to the query. That is, image retrieval methods should be able to bridge the *semantic gap* between the visual appearance of the images and the semantic meaning they convey [26]. But in the specific case of flowchart images, this problem is still ill-defined. Take, for instance, the two flowcharts depicted in Fig. 13.2. It is obvious that the two flowcharts are visually similar, and one could argue that since they are formed by the same subset of symbols and share the same interconnection between them, it is reasonable to consider them as relevant hits in a retrieval scenario. On the other hand, the two flowcharts represent completely different procedures and carry different information, so it is fair to consider them non-relevant one to the other. Since flowcharts carry an important semantic meaning, it would be beneficial to 'translate' such graphical information into a structured format that will allow to browse the contained information, that is, to automatically *understand* and *interpret* flowcharts.

Early works such as [4] and [22] were already focused on the recognition of line drawings for further automatic process. This research line continued until the mid-1990s [3, 50] when most of the research efforts were refocused on the treatment of online sketched drawings [41, 51], although some recent research in those lines can

still be found [45]. To our best knowledge, no commercial patent retrieval system uses graphical interpretation techniques for the retrieval of non-textual information in patent documents. The only efforts in that direction come from the Image Mining for Patent Exploration (**IMPEx**) Project;[3] its main objective is the extraction of semantic information from patent images.

## 13.3  Flowchart Recognition Architecture

Most of the state-of-the-art flowchart recognition architectures follow more or less the same architectures and the same steps. We will overview in this section such steps and the different approaches that have been used in the literature. The first stage is devoted to separate the flowchart image into two different layers, one containing the textual information and the other one containing the graphical elements. Subsequently, graphical elements are separated into two different groups: the symbolic elements (rectangles, circles, ellipses, etc.) and the connectors between them. Afterwards, proper recognition modules are applied in order to recognise which graphical primitives appear at each location and to transcribe the text within the image into electronic format by means of optical character recognition (OCR) techniques. Finally, a structural and syntactic validation step is applied in order to correct the recognition errors by including context-dependent information and eventually producing a structured output describing the flowchart's contents.

Let us overview in the following each of these individual steps. We will finally summarise how the performance of the flowchart recognition systems was evaluated in the context of the CLEF-IP 2012 flowchart recognition task.

### 13.3.1  Text/Graphics Separation

Textual terms appearing within flowcharts cannot be directly recognised following classical OCR approaches that assume a regular layout organised in columns, paragraphs and lines.

The text/graphics separation process aims at segmenting the document into two layers: a layer which contains text characters and annotations and a layer containing graphical objects.

Although there exists a wide taxonomy of text/graphics separation methods [16, 24, 44], the most commonly used are the ones relying on morphological operations and the ones based on connected component analysis.

First, the text/graphics separation methods based on morphological operators assume that the text is what remains after applying iterative openings to the

---

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

original image with structuring elements designed to eliminate rectilinear objects. The method proposed by Wahl et al. [48] is one of the first methods based on morphological filtering. It uses run-length smoothing algorithm (RLSA) to detect vertical and horizontal text strings. RLSA can be seen as morphological closing (or opening) operations with vertical and horizontal structuring elements of length according to the text size and graphical lines width. The method of Lu [25] uses RLSA too. The main improvement of this work is that it allows to detect slanted lines by performing a stretching operation at different angles. The main drawback of these approaches is that they tend to wrongly label text as graphics.

On the other hand, the methods based on connected component analysis are the most commonly used. A pioneer and well-known work was proposed by Fletcher and Kasturi [14]. The basic idea is to segment text based on basic perceptual grouping properties. Thus, simple heuristics on font size, inter-character, word and line spacing and alignment are used. The method requires many thresholds, but the good point is that they are extracted from the image object properties and are not manually set a priori. The main steps of this method are:

1. Connected component generation
2. Filtering of connected components based on area and size
3. Connected component grouping in terms of area and size to cluster those that are likely to belong to the same font size and so are candidates to be in the same string
4. Hough transform applied to the centroids of connected components (text strings are supposed to have a rectilinear arrangement)
5. Logical grouping of strings into words and phrases. This step intends to capture those components kept aside by the Hough step, but that fall into the potential text area (in terms of interline spacing, inter-character gaps, etc.). For example, a period at the end of a string or an accent
6. Text string separation

This method combines simplicity with good performance and scalability to different text properties. This is probably the reason that most of the methods are based on the Fletcher and Kasturi one, with small variations and adaptations to different contexts. The weakness of this method is that it does not cope with text touching graphics. Tombre et al. [44] proposed an improved approach able to separate text touching to graphical parts. In addition they introduced some more heuristics allowing to improve the performance.

The method from Tombre et al. [44] was used in the context of flowchart recognition in [34, 35]. We can see an example of the results produced by the text/graphics separation step in Fig. 13.3.
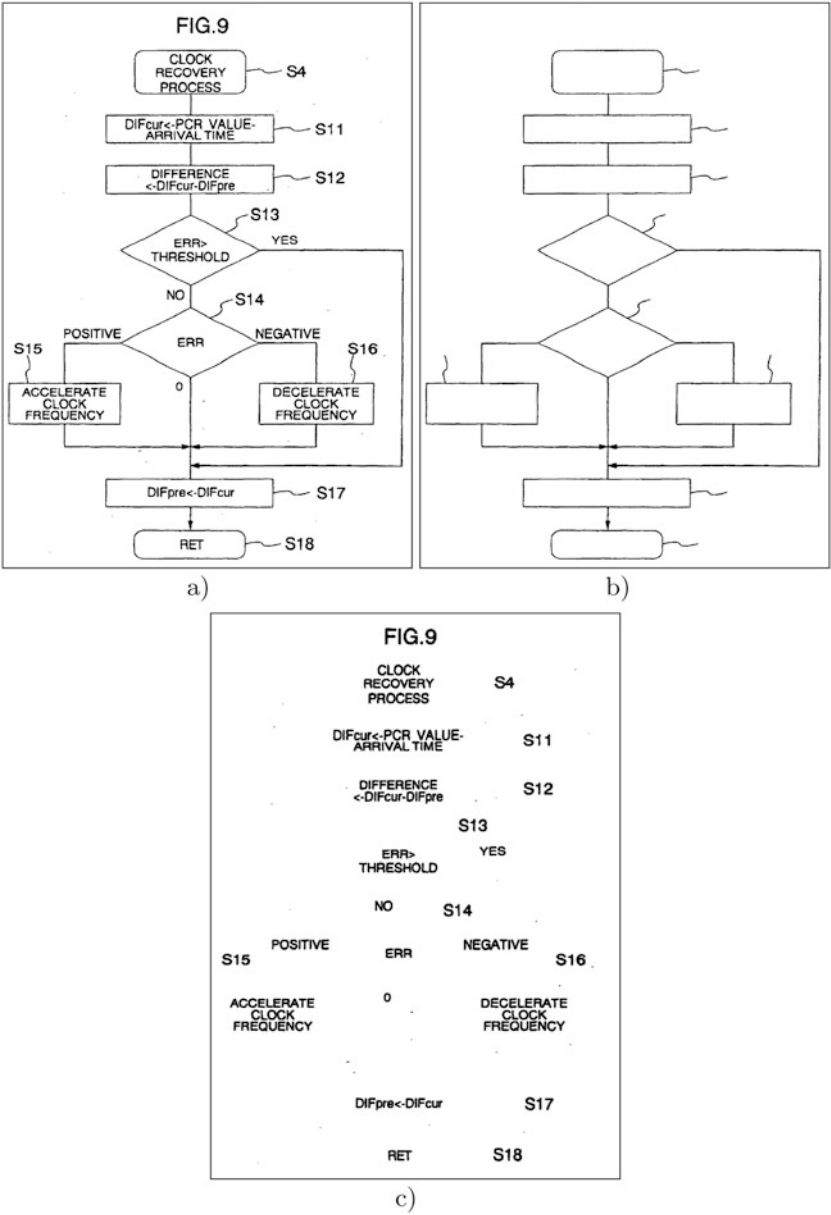
**FIG.9**

CLOCK RECOVERY PROCESS — S4

DIFcur<-PCR VALUE-ARRIVAL TIME — S11

DIFFERENCE <-DIFcur-DIFpre — S12

S13

ERR> THRESHOLD — YES

NO — S14

POSITIVE — ERR — NEGATIVE
S15      0      S16

ACCELERATE CLOCK FREQUENCY

DECELERATE CLOCK FREQUENCY

DIFpre<-DIFcur — S17

RET — S18

a)

b)

**FIG.9**

CLOCK RECOVERY PROCESS      S4

DIFcur<-PCR VALUE-ARRIVAL TIME      S11

DIFFERENCE <-DIFcur-DIFpre      S12

S13

ERR> THRESHOLD      YES

NO      S14

POSITIVE      ERR      NEGATIVE
S15      0      S16

ACCELERATE CLOCK FREQUENCY

DECELERATE CLOCK FREQUENCY

DIFpre<-DIFcur      S17

RET      S18

c)

**Fig. 13.3** Example of the text/graphics separation module. (**a**) Original image, (**b**) graphical layer, (**c**) textual layer, (**d**) undetermined layer (empty in this example)

### 13.3.2 Node and Edge Segmentation

After having separated the text appearing in the flowchart with the graphical entities, most approaches apply a segmentation method in order to separate the nodes (symbols from the flowcharts) rather than the edges (the connectors that define the flow). Here again, a connected component analysis on the graphical layer of the flowchart image drives this segmentation procedure [34, 42]. Closed regions in the flowchart image usually correspond to the nodes of the flowchart. After having determined which regions of the flowchart correspond to nodes, the remaining foreground pixels are attributed as being edges. We can see an example of the node and edge segmentation in Fig. 13.4.

However, this procedure presents two drawbacks. The first is nodes that because of some degradation or some design choice are not fully connected and are likely to be labelled as background zone and be completely missed. This problem is usually tackled by having a pre-processing step that 'closes' the small gaps between co-linear line segments [29, 42]. The second problem is somehow more fundamental. Closed regions in flowcharts do not always correspond to nodes.

Edges linking two non-consecutive nodes in the flow are likely to form loops in the flowchart that will be labelled as a connected component. We can see an example in Fig. 13.5. This case is a clear example of what is known as the Sayre paradox [39]:

> In order to achieve good recognition results, the objects should be previously segmented, but to get reliable segmentation, the objects should be previously recognized.

That is, before running the node recognition module, we need to segment nodes from the background, but to really have a good segmentation without loops appearing as nodes, we need to already define what is a node and what is not. This problem is addressed by casting some heuristics on the shape of the connected components to assess whether the connected component is really a node from the flowchart or corresponds to a loop. For instance, in [35], convexity and vertical symmetry measurements are used in order to discriminate between loops and nodes, since nodes tend to be vertically symmetric and tend also to be convex shapes.

### 13.3.3 Text and Symbol Recognition

After having completely segmented the different regions from the flowchart image comes the proper recognition stage, both for text and for graphical entities.

OCR, being one of the first problems addressed from the pattern recognition field, is considered nowadays an almost solved problem when applied to documents under certain conditions. However, as we have stated before, applying an OCR directly to a flowchart image is likely to fail, since the layout of a flowchart does not follow the same rules as text being printed in a book. But if we feed to an OCR engine the text bounding boxes arisen from the text/graphics separation stage, the results should be acceptable enough. Nowadays, commercial OCR engines such as
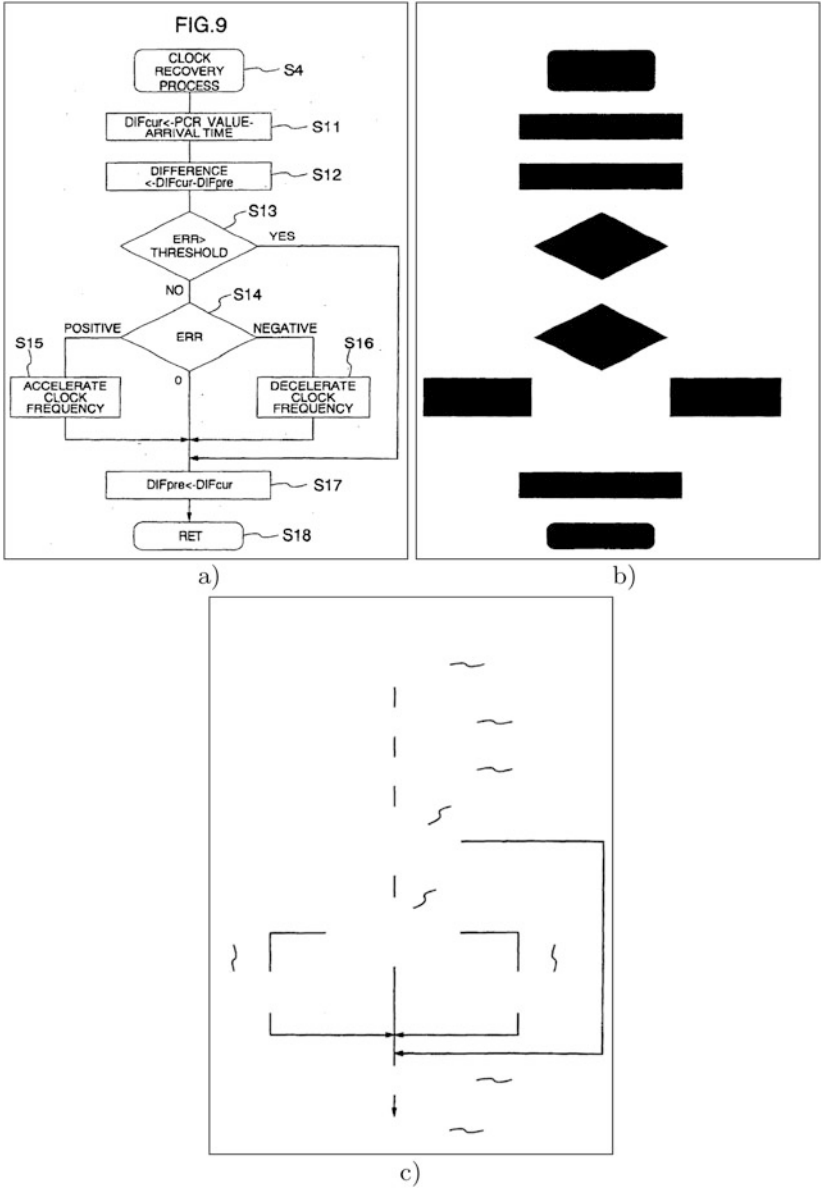
**Fig. 13.4** Example of the node and edge segmentation modules. (**a**) Original image, (**b**) node layer, (**c**) edge layer
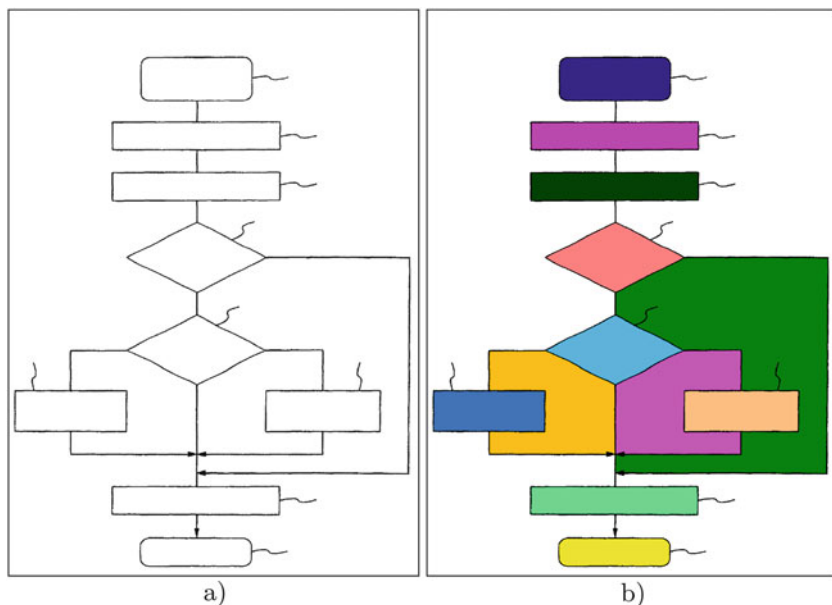
**Fig. 13.5** Example of the connected component labelling, labelling loops as foreground nodes. (**a**) Original input, (**b**) Components identified

ABBYY FineReader,[4] Tesseract[5] and Omnipage[6] are the ones most often used in any document image task. OCR accuracies can be boosted if we provide the OCR engine a context-dependent lexicon and language model as suggested in [42].
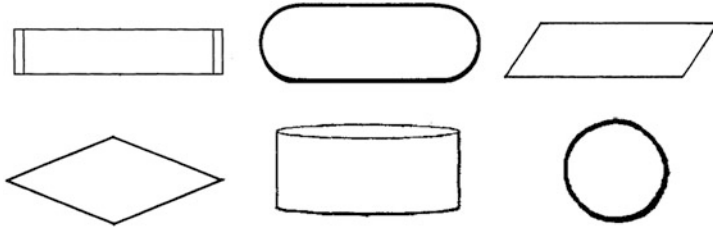
Concerning the recognition of the flowchart's symbols (c.f. Fig. 13.6), in principle any shape descriptor [54] could be used in order to accurately classify the symbols. In [34], Hu geometric moment invariants [53] and the BSM descriptor [9] were used. In both runs submitted to the CLEF-IP 2012 flowchart recognition task from Thean et al. [42] and Mörzinger et al. [29], ad hoc symbol descriptors based on shape symmetry were proposed. From the result analysis of the flowchart recognition task [31, 35], we can see better recognition accuracies were reached when using hand-crafted descriptors for the specific purpose of node recognition than when using generic shape descriptors from the literature.

It is also worth to mention that in [42], a shape normalisation step was proposed in order to deal with the different styles that the same symbol can present (c.f. Figs. 3 and 4 in [42]). A set of squeezing operations result in a shape simplification that helped to improve the node recognition accuracy.

---

[4]http://finereader.abbyy.com/.

[5]http://code.google.com/p/tesseract-ocr/.

[6]http://www.nuance.com/for-individuals/by-product/omnipage/index.htm.

**Fig. 13.6** Example of different node types

## 13.3.4 Structured Output

Once we have identified all the elements of a flowchart, we have to infer from the different relationships among elements which is the structure of the flowchart. More specifically, we have to assess which nodes are connected by an edge. This can be done by simply pairwisely selecting all the detected nodes and subsequently analysing whether any element of the edge layer provokes that those two disjoint nodes merge into a single element. If this happens, then the two nodes are linked through this edge in the delivered graph structure [35].

Subsequently, a structured output of the interpretation of the flowchart has to be provided. Because of the nature of flowcharts, having a graph data structure representation seems to be the most suitable. We can see in Fig. 13.7 an example of the structured output format expected in the CLEF-IP 2012 task [31].

## 13.3.5 Structural and Syntactic Validation

Flowcharts are composed by a set of node symbols together with their connectors and the textual content. However, they also follow a quite strict diagrammatic notation defined by a set of rules that have to be followed so that the flowchart makes sense. None of the CLEF-IP 2012 flowchart recognition task participants used this context knowledge that can yield a strong boost in performance. Graph grammars have been used through the years in order to define a set of rules that two-dimensional signals (i.e. electronic diagrams, flowcharts, architectural drawings, etc.) have to follow in order to be valid [4, 38].

In [20], Lemaitre et al. based their online flowchart recognition system on a structural description with the addition of syntactic knowledge using a grammatical description. We strongly believe that successful flowchart recognition systems should integrate such syntactic definitions in order to reach the desired recognition performances.
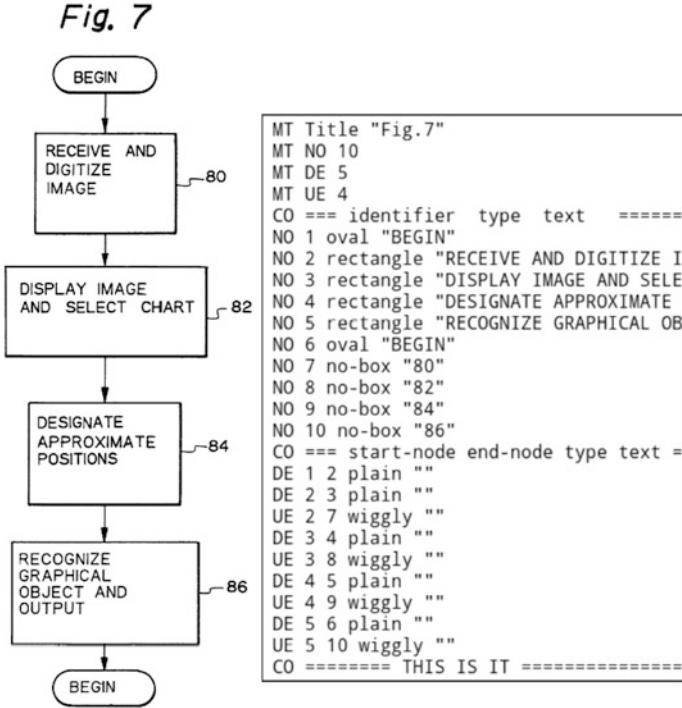
**Fig. 13.7** An example of input image with its corresponding textual information (extracted from [31])

### 13.3.6   *Performance Evaluation of Flowchart Recognition in CLEF-IP 2012*

The flowchart recognition task from the CLEF-IP 2012 campaign was evaluated at three different levels: namely, how well the flowchart structure has been recognised (*structural level*), how well the nodes and the edge types have been recognised (*recognition level*) and a third level that evaluated the text label transcription (*transcription level*).

In order to assess the methods' performance at structural level, a graph metric distance between the topic flowchart $F_t$ and the submitted flowchart $F_s$ is defined in terms of the most common subgraph, mcs$(F_t, F_s)$ [5, 49]. Formally, it was computed as follows:

$$d(F_t, F_s) = 1 - \frac{|\text{mcs}(F_t, F_s)|}{|F_t| + |F_s| - |\text{mcs}(F_t, F_s)|}, \tag{13.1}$$

where $|F_i|$ denotes the size of the graph computed as the number of nodes plus the number of edges.

The most common subgraph measure can be interpreted as follows. When comparing a recognised flowchart $F_s$ and the ground truth expected output $F_t$, the maximum common subgraph $mcs(F_t, F_s)$ measures how well the participant's output matches the expected graph. If the participant's method output is perfect, the maximum common subgraph is the flowchart itself and thus $mcs(F_t, F_s) = |F_t| = |F_s|$ and $d(F_t, F_s) = 0$. If the output is missing a node or some edges, the common structure shared between the output and the ground truth will be smaller than $F_t$, and consequently since $|mcs(F_t, F_s)| < |F_t|$, the final distance $d(F_t, F_s) > 0$ will increase as long as we keep missing elements. The same applies if we deliver an output with extra elements than the ground truth.

The ability to recognise the nodes and the edge types of the different submitted runs was evaluated by the accuracy of the classification, whereas the performance of the textual transcription was measured with a normalised edit distance between the automatically transcribed text and the yielded automatic transcription from the methods.

Although such measures helped to assess which method performs the best at extracting the structure and the contents of the flowchart, it is still unclear if such indicators will exactly correlate with the user experience in a retrieval scenario.

## 13.4  Challenges

Although the recognition of flowcharts has been a present problem for many years in the graphics recognition domain, there are still some challenges that need to be addressed in order to perform patent searches that take into account graphical information. Let us briefly overview which are to our understanding the remaining problems.

– **The Sayre paradox** has to be properly addressed in the flowchart recognition architecture. The whole flowchart recognition relies on an initial segmentation step between nodes, edges, text and background that is far from being perfect. An incorrect segmentation ruins the subsequent recognition steps. It would be thus desirable to have some methods that perform the segmentation and recognition in a single step, as in the case of symbol spotting [33].
– **A syntactic analysis** of flowcharts [20] is a must in order to reach acceptable recognition performances. Syntactic rules should not only serve as a final validation tool, but should drive the whole recognition framework.
– **The inclusion of flowcharts in patent searches** does not end with the flowchart recognition step. Once flowchart images have been 'translated' to a structured format, we would still need a retrieval framework that allows the user to cast queries and to retrieve relevant information stored in graphical format. It is still unclear how such a retrieval system should be designed. Would the user cast their queries using keywords or providing a flowchart sample?

– **The combination of graphical search with textual/semantic information** should be paramount. An effective patent search cannot be just focused on the graphical information, but should be supported by other information queues encoded as metadata in the patent applications. Ideas from different works such as [7, 12] that cast queries covering several granularity levels of information should be adopted in the particular scenario of patent search through flowchart information.

– **The evaluation of flowchart recognition** for the final purpose of information retrieval is also still ill-defined. Which flowcharts can be considered as relevant given a query? Is the fact of missing an edge or misrecognising a node paramount to the final retrieval performance?

## 13.5   Conclusions

In this chapter we have analysed the current technologies available to deal with graphical information in patent retrieval applications and, in particular, with the problem of recognising and understanding information carried by flowcharts. An overview of the different steps that compound a flowchart recognition system has been presented.

Although initiatives such the CLEF-IP 2012 flowchart recognition task mark an important milestone to assess the performance of state-of-the-art methods and track the progress in this specific domain, we have seen that there are still many important challenges yet to be addressed in order to include graphical information in the patent information retrieval framework.

We strongly believe that the use of syntactic knowledge, together with the definition of the retrieval mechanisms dealing with graphical information other than CBIR, is paramount in order to achieve a useful patent graphical information retrieval.

## References

1. Adams S (2005) Electronic non-text material in patent applications – some questions for patent offices, applicants and searchers. World Patent Inf 27(2):99–103
2. Bhatti N, Hanbury A (2013) Image search in patents: a review. Int J Doc Anal Recognit 16(4):309–329
3. Blostein D (1996) General diagram-recognition methodologies. In: Graphics recognition methods and applications. Lecture notes in computer science, vol 1072. Springer, New York, pp 106–122

4. Bunke H (1982) Attributed programmed graph grammars and their application to schematic diagram interpretation. IEEE Trans Pattern Anal Mach Intell 4(6):574–582
5. Bunke H, Shearer K (1998) A graph distance metric based on the maximal common subgraph. Pattern Recogn Lett 19(3–4):255–259
6. Cheng YQ, Cao YL, Yang JY (1990) An automatic recognition system of assembly drawings. In: Workshop on machine vision applications, pp 211—214
7. Codina J, Pianta E, Vrochidis S, Papadopoulos S (2008) Integration of semantic, metadata and image search engines with a text search engine for patent retrieval. In: Proceedings of the workshop on semantic search at the fifth European semantic web conference, pp 14–28
8. Csurka G, Renders JM, Jacquet G (2011) XRCE's participation at patent image classification and image-based patent retrieval tasks of the CLEF-IP 2011. In: CLEF 2011 evaluation labs and workshop, online working notes
9. Escalera S, Fornés A, Pujol O, Radeva P, Sánchez G, Lladós J (2009) Blurred shape model for binary and grey-level symbol recognition. Pattern Recogn Lett 30(15):1424–1433
10. European IPR Helpdesk. How to search for patent information. Fact Sheet, Nov 2011
11. European IPR Helpdesk. Automatic patent analysis. Fact Sheet, Dec 2013
12. Fafalios P, Salampasis M, Tzitzikas Y (2013) Exploratory patent search with faceted search and configurable entity mining. In: Proceedings of the integrating IR technologies for professional search workshop
13. Filippov IV, Nicklaus MC (2009) Extracting chemical structure information: optical structure recognition application. In: Proceedings of the IAPR international workshop on graphics recognition
14. Fletcher LA, Kasturi R (1988) A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans Pattern Anal Mach Intell 10(6):910–918
15. Hanbury A, Bhatti N, Lupu M, Mörzinger R (2011) Patent image retrieval: a survey. In: Proceedings of the fourth workshop on patent information retrieval, pp 3–8
16. Hoang TV, Tabbone S (2010) Text extraction from graphical document images using sparse representation. In: Proceedings of the 9th IAPR international workshop on document analysis systems, pp 143–150
17. Huet B, Kern NJ, Guarascio G, Merialdo B (2001) Relational skeletons for retrieval in patent drawings. In: Proceedings of the international conference on image processing, pp 737–740
18. Kasturi R, Tombre K (1995) Graphics recognition: methods and applications. Lecture notes in computer science, vol 1072. Springer, New York
19. Kesidis A, Karatzas D (2014) Logo and trademark recognition. Handbook of document image processing and recognition. Springer, London, pp 591–646
20. Lemaitre A, Mouchère H, Camillerapp J, Coüasnon B (2013) Interest of syntactic knowledge for on-line flowchart recognition. In: Graphics recognition new trends and challenges. Springer, New York, pp 89–98
21. Lew MS, Sebe N, Djeraba C, Jain R (2006) Content-based multimedia information retrieval: state of the art and challenges. ACM Trans Multimed Comput Commun Appl 2(1):1–19
22. Lin X, Shimotsuji S, Minoh M, Sakai T (1985) Efficient diagram understanding with characteristic pattern detection. Comput Vision Graphics Image Process 30(1):84–106
23. List J (2007) How drawings could enhance retrieval in mechanical and device patent searching. World Patent Inf 29(3):210–218
24. Lladós J, Rusiñol M (2014) Graphics recognition techniques. Handbook of document image processing and recognition. Springer, London, pp 489–521
25. Lu Z (1998) Detection of text regions from digital engineering drawings. IEEE Trans Pattern Anal Mach Intell 20(4):431–439
26. Lupu M, Schuster R, Mörzinger R, Piroi F, Schleser T, Hanbury A (2012) Patent images – a glass-encased tool: opening the case. In: Proceedings of the 12th international conference on knowledge management and knowledge technologies
27. Mahmoudi F, Shanbehzadeh J, Eftekhari-Moghadam AM, Soltanian-Zadeh H (2003) Image retrieval based on shape similarity by edge orientation autocorrelogram. Pattern Recogn 36(8):1725–1736

28. Mörzinger R, Horti A, Thallinger G, Bhatti N, Hanbury A (2011) Classifying patent images. In: CLEF 2011 evaluation labs and workshop, online working notes
29. Mörzinger R, Schuster R, Horti A, Thallinger G (2012) Visual structure analysis of flow charts in patent images. In: CLEF 2012 evaluation labs and workshop, online working notes
30. Piroi F, Lupu M, Hanbury A, Zenz V (2011) CLEF-IP 2011: retrieval in the intellectual property domain. In: CLEF 2011 evaluation labs and workshop, online working notes
31. Piroi F, Lupu M, Hanbury A, Sexton AP, Magdy W, Filippov IV (2012) CLEF-IP 2012: retrieval experiments in the intellectual property domain. In: CLEF 2012 evaluation labs and workshop, online working notes
32. Rusiñol M, Lladós J (2010) Efficient logo retrieval through hashing shape context descriptors. In: Proceedings of the ninth IAPR international workshop on document analysis systems, pp 215–222
33. Rusiñol M, Lladós J (2010) Symbol spotting in digital libraries: focused retrieval over graphic-rich document collections. Springer, London
34. Rusiñol M, de las Heras LP, Mas J, Terrades OR, Karatzas D, Dutta A, Sánchez G, Lladós J (2012) CVC-UAB's participation in the flowchart recognition task of CLEF-IP 2012. In: CLEF 2012 evaluation labs and workshop, online working notes
35. Rusiñol M, de las Heras LP, Ramos O (2014) Flowchart recognition for non-textual information retrieval in patent search. Inf Retr 17(4):331–341
36. Sadawi N (2009) Recognising chemical formulas from molecule depictions. In: Proceedings of the IAPR international workshop on graphics recognition
37. Samet H, Webber RE (1985) Storing a collection of polygons using quadtrees. ACM Trans Graph 4(3):182–222
38. Sánchez G, Lladós J (2004) Syntactic models to represent perceptually regular repetitive patterns in graphic documents. In: Graphics recognition. Recent advances and perspectives. Springer, New York, pp 166–175
39. Sayre KM (1973) Machine recognition of handwritten words: a project report. Pattern Recogn 5(3):213–228
40. Sidiropoulos P, Vrochidis S, Kompatsiaris I (2011) Content-based binary image retrieval using the adaptive hierarchical density histogram. Pattern Recogn 44(4):739–750
41. Szwoch W (2007) Recognition, understanding and aestheticization of freehand drawing flowcharts. In: Proceedings of the ninth international conference on document analysis and recognition, pp 1138–1142
42. Thean A, Deltorn JM, Lopez P, Romary L (2012) Textual summarisation of flowcharts in patent drawings for CLEF-IP2012. In: CLEF 2012 evaluation labs and workshop, online working notes
43. Tiwari A, Bansal V (2004) PATSEEK: content based image retrieval system for patent database. In: Proceedings of the fourth international conference on electronic business, pp 1167–1171
44. Tombre K, Tabbone S, Pelissier L, Lamiroy B, Dosch P (2002) Text/graphics separation revisited. In: Document analysis systems V. Lecture notes in computer science, vol 2423. Springer, New York, pp 615–620
45. Vasudevan BG, Dhanapanichkul S, Balakrishnan R (2008) Flowchart knowledge extraction on image processing. In: Proceedings of the IEEE international joint conference on neural networks, pp 4075–4082
46. Vrochidis S, Papadopoulos S, Moumtzidou A, Sidiropoulos P, Pianta E, Kompatsiaris I (2010) Towards content-based patent image retrieval: a framework perspective. World Patent Inf 32(2):94–106
47. Vrochidis S, Moumtzidou A, Kompatsiaris I (2012) Concept-based patent image retrieval. World Patent Inf 34(4):292–303
48. Wahl F, Wong K, Casey R (1982) Block segmentation and text extraction in mixed text/image documents. Comput Graphics Image Process 20(4):375–390
49. Wallis WD, Shoubridge P, Kraetz M, Ray D (2001) Graph distances using graph union. Pattern Recogn Lett 22(6–7):701–704

50. Yu Y, Samal A, Seth SC (1997) A system for recognizing a large class of engineering drawings. IEEE Trans Pattern Anal Mach Intell 19(8):868–890
51. Yuan Z, Pan H, Zhang L (2008) A novel pen-based flowchart recognition system for programming teaching. In: Advances in blended learning. Lecture notes in computer science, vol 5328. Springer, Berlin, pp 55–64
52. Zesheng S, Jing Y, Chunhong J, Yonggui W (1994) Symbol recognition in electronic diagrams using decision tree. In: Proceedings of the IEEE international conference on industrial technology, pp 719–723
53. Zhang D, Lu G (2002) A comparative study of three region shape descriptors. In: Proceedings of the digital image computing techniques and applications, pp 1–6
54. Zhang D, Lu G (2004) Review of shape representation and description techniques. Pattern Recogn 37:1–19