K. C. Santosh

Document Image Analysis

Current Trends and Challenges in Graphics Recognition

Springer

Document Image Analysis

K. C. Santosh

Document Image Analysis

Current Trends and Challenges in Graphics Recognition

Springer

K. C. Santosh
Department of Computer Science
University of South Dakota
Vermillion, SD, USA

ISBN 978-981-13-2338-6 ISBN 978-981-13-2339-3 (eBook)
https://doi.org/10.1007/978-981-13-2339-3

Library of Congress Control Number: 2018952602

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

I dedicate this book to my wife Dhoju, Anju, who has been motivating for several years whether I can write a book in the domain as she knew that I wrote several peer-reviewed research articles, proceedings, and chapters. The area I am confident on and have more than a decade of experience is pattern recognition, and the topic I chose for a book is graphics recognition. This book revisits the works reported for graphics recognition since the 90s, and the primary source is my Ph.D. thesis that was supervised by two brilliant professors: Wendling, Laurent and Lamiroy, Bart—many thanks to them (a big salute). How can I forget to dedicate to my mother K.C., Jamuna and late father K.C., Lokesh? I strongly believe my son, Aarush (of 5), will continue to serve the academic world.

Foreword

I am, personally, honored to have this opportunity to write a few words (as foreword) for the book entitled *Document Image Analysis: Current Trends and Challenges in Graphics Recognition*. About me, I have several years of experience in the pattern recognition domain, especially in graphics recognition, and I worked as the President of the Graphics Recognition (GREC) Technical Committee-10 (TC-10), which is in the framework of the International Association for Pattern Recognition (IAPR).

The book starts with a clear and concise overview of document image analysis; the author puts a position about where does graphics processing lie (Chap. 1), which is immediately followed by graphics recognition (Chap. 2) in detail. The best part of the book is it summarizes the rich state-of-the-art techniques in addition to those international contests that have been happening in every 2 years since the 90s. This summary helps readers understand the scope and importance of graphics recognition in the domain. Another important issue is the author framed the need for validation protocol (Chap. 3) so that it allows a fair comparison that let us review our advancements then and now. Three different fundamental approaches, viz. statistical (Chap. 4), structural (Chap. 5), and syntactic (Chap. 7), are comprehensively described for graphics recognition by taking state-of-the-art (up to date) research techniques in addition to the hybrid approaches (Chap. 6). For a complex graphics recognition problem, structural approaches are found to be appropriate and have been well covered in the book. Interestingly, even though there exist a few works on the syntactic approach for graphical symbol recognition, the author sets a position and its importance as the image description happens to be close to human understanding language. The summary of the book (Chap. 8) is succinct and to the point, which is the best in the book. One of the primary reasons behind this is graphics recognition is not just limited to document imaging problems, such as architectural drawings, electrical circuit diagrams, and maps, but also (bio)medical imaging. This clearly opens the space for graphics recognition community and help researchers move forward, and at the same time, other researchers (outside the graphics recognition community) can take advantage of graphics recognition state-of-the-art techniques. Therefore, I strongly believe the book has the potential to attract a large audience.

La Rochelle, France
July 2018

Jean-Marc Ogier, Ph.D.
President
University of La Rochelle

Preface

The book focuses on a challenging research topic: graphics recognition—a subfield of a larger research domain, i.e., pattern recognition—which has been considered as a key problem toward document content understanding and interpretation, and mostly architectural and engineering drawings and electrical circuit diagrams. In general, starting with its definition, we have discussed basic steps used in state-of-the-art methods, major applications, research standpoints based on several dedicated methods for graphics recognition. In the 60s and 70s, the resource-constrained machines did not allow the use of complex recognition techniques [1] and few data were processed. Since then, the development of more powerful computer machines, interactions between disciplines, and the introduction of new applications (data mining, creating a taxonomy of documents) led to the development of several concepts [2]. Graphics recognition has had an extremely rich state-of-the-art literature in symbol recognition and localization since the 70s [3], where the state-of-art methods are categorized into three approaches: statistical, structural, and syntactic.

As stated before, the book covers *statistical*, *structural*, and *syntactic* approaches and addresses their merits and demerits considering the context. Through comprehensive experiments, it also provides an idea of whether the aforementioned approaches can be combined. It, in general, contains research problems, and state-of-the-art methods that convey basic steps as well as prominent techniques, evaluation metrics and protocols, and research standpoints/directions that are associated with it. The book is not limited to straightforward isolated graphics (visual patterns) recognition. It aims to address complex and composite graphical symbol recognition [4–12]. Recent trends on several other different (but major) real-world problems are also discussed. Further, few examples will demonstrate to see whether graphics recognition can be extended to other domains, such as graphics recognition in medical imaging, so that we can prove that graphics recognition is not just limited to a few problems, such as technical drawings, architectural drawings, electrical circuit diagrams, and other document imaging.

Vermillion, USA
July 2018

K. C. Santosh, Ph.D.
Assistant Professor and Graduate
Program Coordinator

References

1. G. Nagy, State of the art in pattern recognition. Proc. IEEE **56**(5), 836–862 (1968)
2. A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
3. M. Rusiñol, J. Lladós, *Symbol Spotting in Digital Libraries: Focused Retrieval over Graphic rich Document Collections* (Springer, London, 2010)
4. D. Doermann, K. Tombre, *Handbook of Document Image Processing and Recognition* (Springer-Verlag New York Incorporated, New York, 2014)
5. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme (Graphics Recognition using Spatial Relations and Shape Analysis). Ph.D. thesis, University of Lorraine, France, 2011
6. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in *Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers,* eds. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi Communications in Computer and Information Science, vol. 709 (2017), pp. 3–15
7. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, New York, 2015), pp. 1–22
8. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: Bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. and Artif. Intell. **28**(06), 1450017 (2014)
9. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
10. K.C. Santosh, B. Lamiroy, L.Wendling. DTW-radon-based shape descriptor for pattern recognition. Int. J. Pattern Recognit. Artificial Intell. **27**(3), 1350008 (2013)
11. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
12. K.C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2009), pp. 1330–1334

Acknowledgements

Since the book is motivated by the Ph.D. report (2011) entitled *Graphics Recognition Using Spatial Relations and Shape Analysis*, I would like to acknowledge my brilliant supervisors: *Wendling, Laurent* (Universitè Paris Descartes (Paris V), France) and *Lamiroy, Bart* (Universitè de Lorraine, France). The Ph.D. work was completed in INRIA Nancy—Grand Est Research Centre and was affiliated with Institut National Polytechnique de Lorraine (INPL), Nancy, and Université de Lorraine (UDL), France.

Contents

About the Author

Dr. K. C. Santosh (Senior Member, IEEE) is currently an assistant professor and graduate program coordinator in the Department of Computer Science at the University of South Dakota (USD). Before joining the USD, he worked as a research fellow at the US National Library of Medicine (NLM), National Institutes of Health (NIH). He worked as a postdoctoral research scientist at the LORIA research centre, Universite de Lorraine, in direct collaboration with the industrial partner ITESOFT, France. He also worked as a research scientist at the INRIA Nancy—Grand Est Research Centre, France, where he completed his Ph.D. in computer science. Before that, he worked as a graduate research scholar at SIIT, Thammasat University, Thailand, where he completed his MS in computer science.

He has demonstrated expertise in pattern recognition, image processing, computer vision, and machine learning with several different applications, such as handwriting recognition, graphics recognition, document information content exploitation, (bio)medical image analysis, biometrics, healthcare informatics and medicine, and IoT. He has published more than 100 peer-reviewed research articles and has given more than 20 invited talks at various universities, international conferences, and symposiums (in and outside the USA). He has edited several books and journal issues, e.g., for the Communications in Computer and Information Science (CCIS) Series, International Journal of Speech Technology, Multimedia Tools and Applications, and International Journal of Healthcare Informatics (all published by Springer). He serves or has served as an editorial board member for several journals (published by IEEE, Springer, and ACM). He has more than 7 years of teaching experience in different capacities—such as teaching assistant, assistant lecturer, head of the department, chief of research and development, and assistant professor—gathered on three different continents: Asia, Europe, and America. For more information, see http://kc-santosh.org.

Acronyms

AAM	Active appearance model
ACE	Automatic content extraction
AI	Artificial intelligence
ALC	Average-linkage clustering
ANN	Artificial neural network
ARG	Attributed relational graph
ARL	Association rule learning
ART	Angular radial transform
BN	Bayesian network
BOR	Bag-of-relations
BOS	Bag-of-symbols
BSM	Blurred Shape Model
BWI	Build-Weed-Incorporate
CAD	Computer-aided design
CCC	Cophenetic correlation coefficient
CLC	Complete-linkage clustering
DAE	Document analysis and exploitation
DB	Davies–Bouldin index
DIA	Document image analysis
DTL	Decision tree learning
DTW	Dynamic time warping
DU	Dunn index
ED	Electrical diagram
FD	Fourier descriptor
FMGE	Fuzzy multilevel graph embedding
FOL	First-order logic
FT	Fourier transform
GFD	Generic Fourier descriptor
GP	Genetic programming
GREC	Graphics recognition

HRT	Histograms of the Radon transform
HT	Hough transform
IAPR	International Association of Pattern Recognition
ILP	Inductive logic programming
KDM	Kernel density matching
k-NN	k-Nearest neighbor
LHRT	Logarithmic of HRT
LP	Logic programming
LSH	Locality-sensitive hashing
MBR	Minimum boundary rectangle
ML	Machine learning
MU	Message Understanding
NER	Named-entity recognition
NLP	Natural language processing
OCR	Optical character recognition
POC	Phase-only correlation
RAG	Region adjacency graph
RANSAC	Random Sample Consensus minimization
RL	Reinforcement learning
ROI	Region-of-interest
RT	Radon transform
SC	Shape context
SF	Score function
SI	Silhouette index
SIFT	Scale-invariant feature transform
SIHA	Statistical integration of histogram array
SLC	Single-linkage clustering
SPR	Syntactic pattern recognition
SVM	Support vector machine
TC-10	Technical Committee-10

Chapter 1
Document Image Analysis

1.1 Document Image Analysis (DIA)

1.1.1 What is Document Imaging?

Conventionally speaking, paper documents are referred to as document images, and are considered for document image analysis/processing. Paper documents have been overwhelmingly increased since a few decades and the we are required to automatically process them. In the early 80s, since printed documents have been widely used, paper-less-office could be the objective of the smart office, where computers needed process them as if experts were used.

In document image analysis/processing, paper documents are scanned and stored for further processing. How about those camera-based documents and those dynamic texts, i.e. running texts in the videos? Can we then define document image processing as scanning-storing-retrieving-managing? In addition, an expected outcome will be limited to producing compatible electronic format so that they are easier to access. Can we just limited to a set of simple techniques and procedures that are used to convert document images (often, scanned) from pixel information into a format that can be read by a computer?

In a broader understanding, document imaging can be taken as the practice of using high-end equipments, such as scanners and cameras so that they can automatically be accessed and processed as expected. Commercially speaking, in what follows, a few things can be itemized that help us understand the meaning of document imaging:

(a) Frequent document information retrieval;
(b) Document sharing;
(c) Document search, indexing and retrieval; and
(d) Special compliance requirements.

In short, it can provide a way to convert the document to a format that can be accessed easily, and of course, it reduces costs and increases efficiency at work. If the invoices

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_1

are imaged, it can be viewed from anywhere. Having the imaged data, text mining can be done and therefore, search and retrieval of similar documents can easily be processed.

Even today, we are surrounded by paper documents, and are often scanned. This means that DIA has a long and significant importance. For example, business forms, postal documents, bank checks, newspapers (media), administration documents that can be from any institutions/agencies, such as hospital, government education and businesses. How good it could be if a machine can automatically process them in accordance with the users need [1]. As a quick note, As reported in [2], the cost of manual data extraction can go up to 9 EUR for a single invoice. At this point, we have to think of a commercial system where thousands of data have to be processed in every single day. Can we stay as it is: manual data extraction, even today? In 2015, document information extraction tool has been proposed and made it available (in collaboration with ITESOFT (https://www.itesoft.com). France) [3–6]. The main idea behind it is to carry out (extract) those information which users think is/are important, not all information, since full-page information can overwhelm the users. Such a tool definitely requires a comprehensive procedures starting from document pre-processing, graphics processing to text processing. An example is illustrated in Fig. 1.1. In this example, it is clear that no other information is/are of interest except those text in blue. This means that table processing does not look trivial since all contents may not be useful. However, under the purview of the DIA, we cannot avoid working on graphics processing, such as logo and line. In short, graphics can ease text processing. In what follows, a set of such major processes will be broadly explained.

1.1.2 Basics to DIA and Challenges

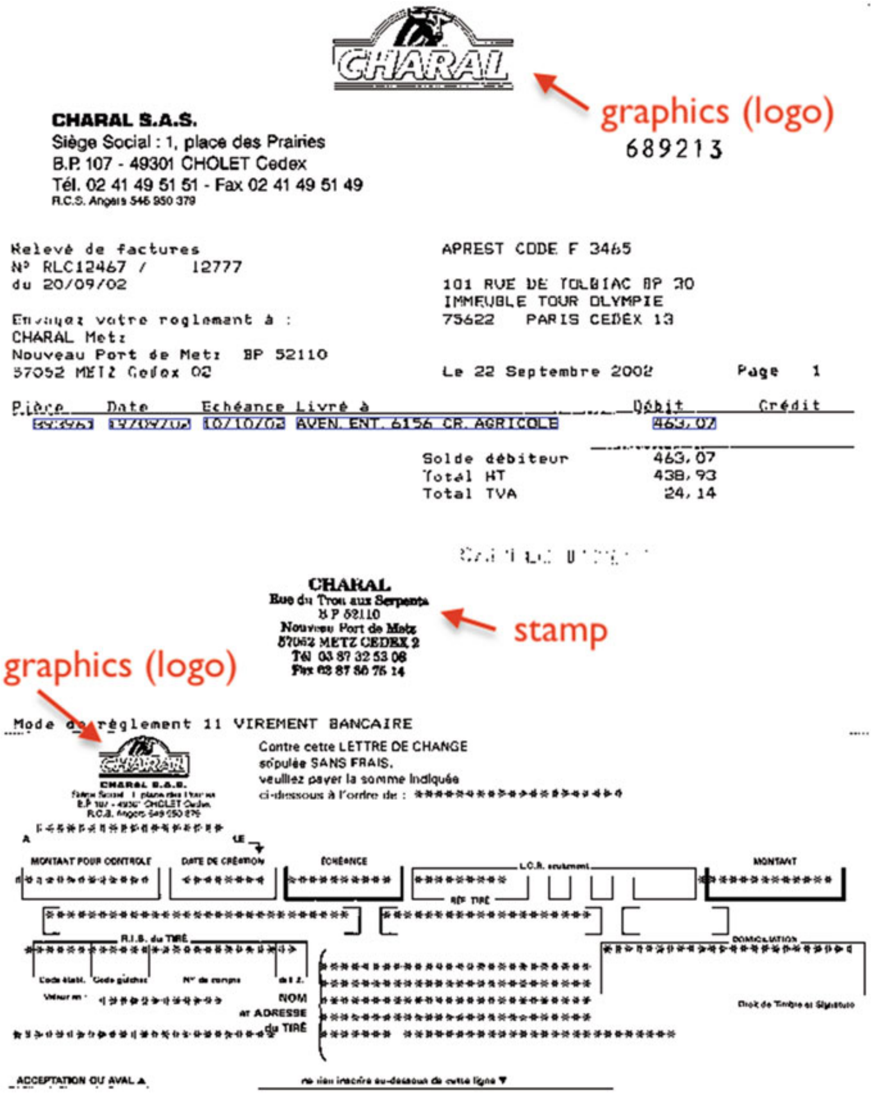
Let us start with the objective of the DIA [1]:

> *The objective of document image analysis is to recognize the text and graphics components in images, and to extract the intended information as a human would.*

This means that DIA is mainly related to texts and graphics, and both could be handwritten and machine-printed. Theses are often related to processes, such as text and or graphics separation, localization (spotting, for instance), and recognition and retrieval [7]. According to [8], DIA can be considered as document image interpretation and understanding. In both research articles [1, 8], core ideas remain the same.

Considering global processes, let us categorize DIA into two domains [1]:

(a) text processing and
(b) graphics processing.



graphics (logo)
stamp
graphics (logo)

CHARAL S.A.S.
Siège Social : 1, place des Prairies
B.P. 107 - 49301 CHOLET Cedex
Tél. 02 41 49 51 51 - Fax 02 41 49 51 49
R.C.S. Angers 545 950 379

689213

Relevé de factures
N° RLC12467 / 12777
du 20/09/02

APREST CODE F 3465

101 RUE DE TOLBIAC BP 30
IMMEUBLE TOUR OLYMPIE
75622 PARIS CEDEX 13

Envoyez votre roglemant à :
CHARAL Metz
Nouveau Port de Metz BP 52110
57052 METZ Cedex 02

Le 22 Septembre 2002 Page 1

Pièce	Date	Echéance	Livré à	Débit	Crédit
893963	19/09/02	10/10/02	AVEN. ENT. 6156 CR. AGRICOLE	463,07	

Solde débiteur	463,07
Total HT	438,93
Total TVA	24,14

CHARAL
Rue du Trou aux Serpents
B P 52110
Nouveau Port de Metz
57052 METZ CEDEX 2
Tél 03 87 32 53 08
Fax 03 87 80 75 14

Mode de réglement 11 VIREMENT BANCAIRE

CHARAL S.A.S.
Siège Social 1 place des Prairies
B.P. 107 - 49301 CHOLET Cedex
R.C.S. Angers 545 950 379

Contre cette LETTRE DE CHANGE
stipulée SANS FRAIS,
veuillez payer la somme indiquée
ci-dessous à l'ordre de :

A LE

MONTANT POUR CONTROLE	DATE DE CRÉATION	ÉCHÉANCE	L.C.R. seulement	MONTANT

REF. TIRÉ

R.I.B. du TIRÉ

Code établ.	Code guichet	N° de compte	clé R.I.B.

DOMICILIATION

Valeur en :

NOM
et ADRESSE
du TIRÉ

Droit de Timbre et Signature

ACCEPTATION OU AVAL ▲

ne rien inscrire au-dessous de cette ligne ▼

Fig. 1.1 Document page: a sample showing a typical table processing problem along with document layout analysis. Tables can happen everywhere, such as header, body and footer of the document page. In the body, annotated texts/information (in blue) are the ones experts/users think is important

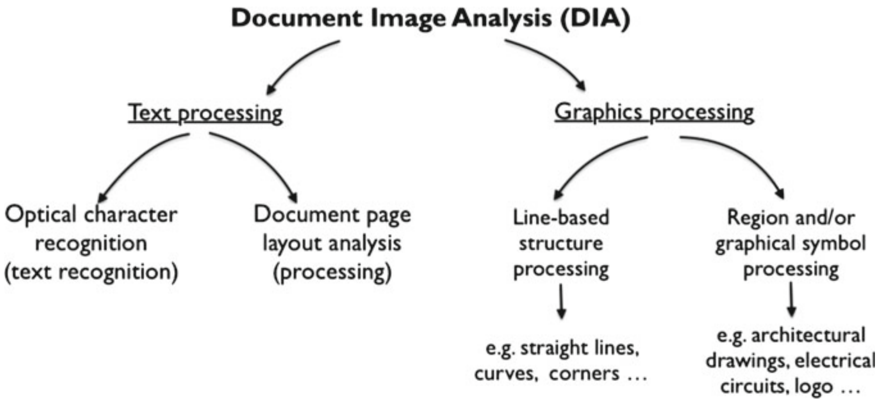

Document Image Analysis (DIA)
Text processing
Graphics processing
Optical character recognition (text recognition)
Document page layout analysis (processing)
Line-based structure processing
Region and/or graphical symbol processing
e.g. straight lines, curves, corners …
e.g. architectural drawings, electrical circuits, logo …

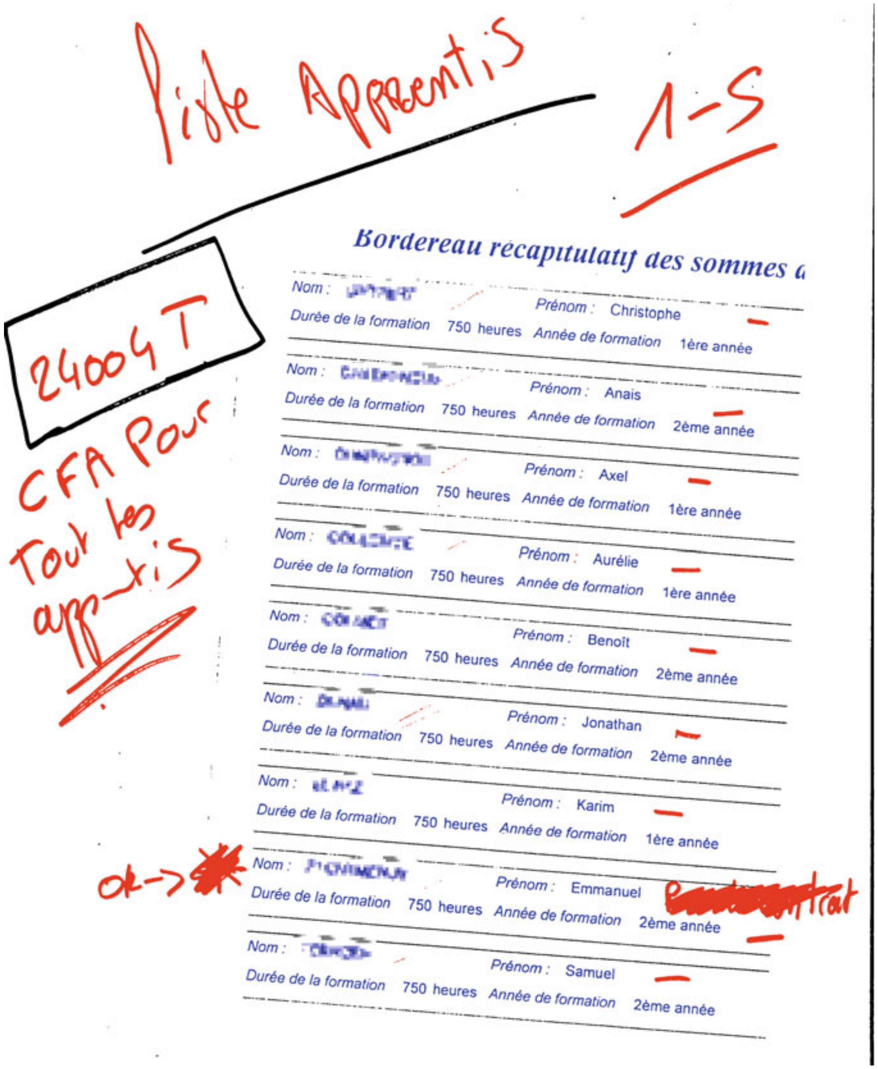
Fig. 1.2 Document image analysis or processing: basic hierarchy

For better understanding, in Fig. 1.2, fundamental document image analysis/processing hierarchy has been clearly illustrated. Note that it does not refer to processes/techniques. In text processing, basic tasks include (and not limited to)

(a) Document skew angle estimation/correction;
(b) Document layout analysis (finding columns, paragraphs, text lines and words);
(c) Handwritten, machine-printed text and graphics separation;
(d) Table detection and processing; and
(e) Optical Character Recognition (OCR) and text recognition (handwritten and machine-printed).

Understanding document's layout can ease subsequent processes [9–13]. Simply, layout understanding and/or analysis refers the division of page into text blocks, lines and words in accordance with the reading order, and it is required to address large scale projects in the domain: document digitization. In general, and in the way Shafait [13] mentioned, a set of few but major processes can be enumerated as follows:

(i) Binarization (a process to convert grayscale or color document image into a bi-level representation);
(ii) Noise removal (a process to detect and remove noise in a document that are usually introduced by processes, such as scanning or binarization);
(iii) Skew correction (a process to detect and correct the document's orientation angle from the horizontal direction);
(iv) Page and zone segmentation (processes to divide a document page into homogeneous zones, where each zone is composed of a set of predefined classes, such as text, image and graphics).



Piste Apprentis

1-S

24004T

CFA Pour Tout les apptis

ok->

Bordereau récapitulatif des sommes d

Nom : [redacted] Prénom : Christophe
Durée de la formation 750 heures Année de formation 1ère année

Nom : [redacted] Prénom : Anais
Durée de la formation 750 heures Année de formation 2ème année

Nom : [redacted] Prénom : Axel
Durée de la formation 750 heures Année de formation 1ère année

Nom : [redacted] Prénom : Aurélie
Durée de la formation 750 heures Année de formation 1ère année

Nom : [redacted] Prénom : Benoît
Durée de la formation 750 heures Année de formation 2ème année

Nom : [redacted] Prénom : Jonathan
Durée de la formation 750 heures Année de formation 2ème année

Nom : [redacted] Prénom : Karim
Durée de la formation 750 heures Année de formation 1ère année

Nom : [redacted] Prénom : Emmanuel trat
Durée de la formation 750 heures Année de formation 2ème année

Nom : [redacted] Prénom : Samuel
Durée de la formation 750 heures Année de formation 2ème année

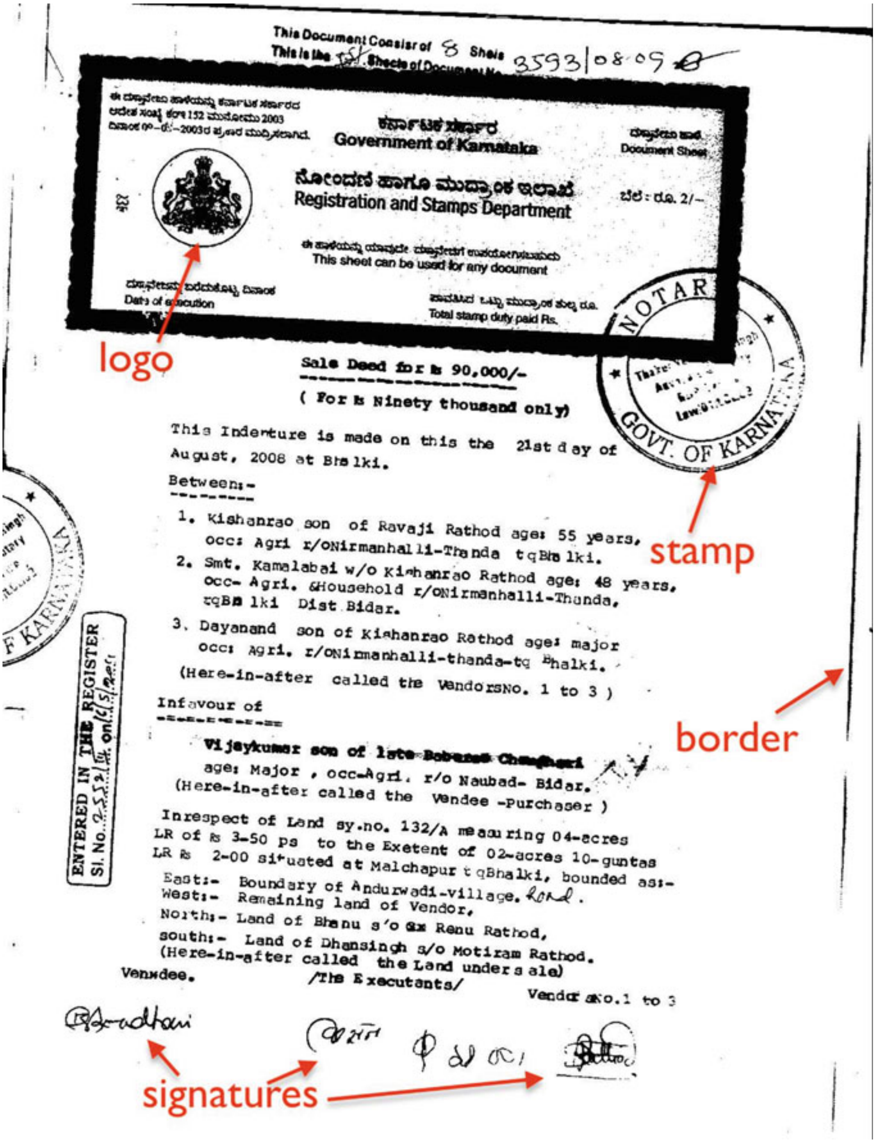
Fig. 1.3 Document page: a sample showing a typical problem of handwritten (in red) and printed (in blue) text separation, including graphics or possible noise (in black)

Note that, the overall idea is to process/arrange document in a way that it is readable. Like we have discussed before, document page layout analysis is not trivial since one needs to deal with possible skewness [14]. Besides, handwritten and machine-printed document page separation in addition the possible use of graphics could be interesting so that further processing is feasible, such as document information retrieval [15, 16]. An example is provided in Fig. 1.3. In [15], authors mentioned that one has to face everyday a varieties of document types, content, quality and structure.

Not limited to these, documents can be skewed, noisy and sometimes overlapped with graphics (unconstrained annotations). It is also important to know such annotations in any languages they wrote, such as French, German and English; to separate the content: typed or handwritten, and to deal with document structure: structured (tables), semi- structured (forms) and possible heterogenous document page. At the same point, we note that table contains a large portion of the information of the documents. For example, invoices, bank transactions and receipts are often happened in tabular format, and they are important as well as sensitive. Connecting the table information with the ones in the paragraph using table caption could be one of the better ideas to narrow down the document information search and retrieval process. Further, end-users may not be interested in header and footer information since they can be repeated for all documents of a specific document category. A comprehensive study has been reported in [3, 5, 6]. In such a process, it could be more efficient, if we are able to identify what type of document (meaning, category or brand, such as 'Nike', 'Adidas' and 'McDonald') is used before we process entire document page. Traditionally, it could be done by detecting and recognizing the 'logo' or any other graphical symbol [17–19] that has been printed in the document page.[1] An example is provided in Fig. 1.4. This is not just limited to demonstrating logo problem but also other important issues, such as handwritten, printed text separation (including graphics), logo detection/recognition (not to be confused with stamp), script identification (in multi-script document page) [20–22], document layout analysis [9–13], border removal [23–25] and signature recognition/verification [26–28]. In Figs. 1.3 and 1.4, we have seen two different samples, where in both cases document clean-up and/or border removal is required. Further processes, such as logo detection/verification and signature (including handwriting) recognition/verification are other important issues. Note that stamp and logo look similar but, they need to be differently treated (see Fig. 1.4). Borders exist when a page has been scanned or photocopied. Textual noise, such as extraneous symbols from the neighboring page of the book and/or non-textual noise, such as black borders, speckles are appeared. Automated signature verification is required in case one has to deal with large data, and it happens more often in bank cheques (checks) processing across the world for any sort of financial transactions. This means that, application is not limited to what has been shown in Fig. 1.4.

It is important to remind that the text processing is not the aim of the book. However, we understand that text processing can not be sidelined from the DIA perspective. In what follows, graphics processing will broadly be explained.

[1] Detecting a graphical symbol 'logo' can be used to provide an example to avoid complexity document retrieval process. In document understanding, there exist several ways to do exactly similar task: document identification. For more information, follow Sect. 1.2.



This Document Consisr of 8 Sheis
This is the 1st Sheets of Document No. 3593/08-09

ಕರ್ನಾಟಕ ಸರ್ಕಾರ
Government of Karnataka

ದಸ್ತಾವೇಜು ಹಾಳೆ
Document Sheet

ನೋಂದಣಿ ಹಾಗೂ ಮುದ್ರಾಂಕ ಇಲಾಖೆ
Registration and Stamps Department

ಬೆಲೆ: ರೂ. 2/-

This sheet can be used for any document

Date of execution

Total stamp duty paid Rs.

NOTARY
GOVT. OF KARNATAKA

Sale Deed for ₨ 90,000/-

(For ₨ Ninety thousand only)

This Indenture is made on this the 21st day of August, 2008 at Bhalki.

Between:-

1. Kishanrao son of Ravaji Rathod age: 55 years, occ: Agri r/oNirmanhalli-Thanda tqBhalki.
2. Smt. Kamalabai w/o Kishanrao Rathod age: 48 years, occ- Agri. &Household r/oNirmanhalli-Thanda, tqBhalki Dist Bidar.
3. Dayanand son of Kishanrao Rathod age: major occ: Agri. r/oNirmanhalli-thanda-tq Bhalki.

(Here-in-after called the VandorsNo. 1 to 3)

Infavour of

Vijaykumar son of late Baburao Chaudhari
age: Major , occ-Agri. r/o Naubad- Bidar.
(Here-in-after called the Vendee -Purchaser)

Inrespect of Land sy.no. 132/A measuring 04-acres LR of ₨ 3-50 ps to the Exetent of 02-acres 10-guntas LR ₨ 2-00 situated at Malchapur tqBhalki, bounded as:-

East:- Boundary of Andurwadi-village. Road.
West:- Remaining land of Vendor,
North:- Land of Bhanu s'o Renu Rathod,
south:- Land of Dhansingh s/o Motiram Rathod.
(Here-in-after called the Land under sale)

Venndee. /The Executants/ Vendor No.1 to 3

ENTERED IN THE REGISTER
Sl. No. 2552/14. on 16/5/2011

Fig. 1.4 Document page: a sample showing several different problems in the DIA, such as handwritten, printed text separation (including graphics), logo detection/recognition (not to be confused with stamp), script identification (in multi-script document page), document layout analysis, border removal and signature recognition/verification

1.2 Graphics Processing

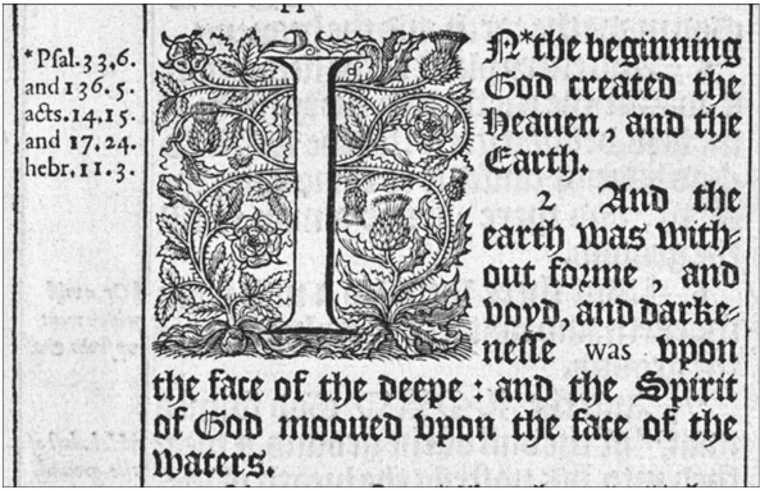
Can we work on text processing without dealing with graphics? Even though, we have clear-cut separation of text and graphics processing (see Fig. 1.2), one cannot simply work on one side of the DIA since a document contains both of them. Where does the graphics processing lie then?

As mentioned earlier, let us start with a few important applications, where document image retrieval, classification and recognition have been greatly influenced by the appropriate use of graphics processing. In our advanced technology, where business data increases everyday. Having massive data in hand, one needs to able to use salient features/entities, such as logos, stamps or seals. Use of logo and stamp detection does not only help document authentication but also useful document retrieval and classification. More detailed information were reported in [17–19]. On the whole, for a massive administrative data, detecting graphics i.e. logo, for instance can help reduce the processing time or speed up the document retrieval and/or classification. This means that graphics processing cannot be sidelined from text processing (Fig. 1.5).

In a similar fashion, text (handwritten and machine-printed) recognition has been significantly merged/influenced by the use of graphics. In this framework, just to name a few, one cannot forget two applications:

(i) drop-cap processing in historical/ancient documents; and
(ii) map processing.

While taking cultural heritage preservation into account, several digitization projects (in Europe, France in particular) to save the contents of thousands of ancient



*Pſal.33,6.
and 136.5.
acts.14.15.
and 17.24.
hebr.11.3.

IN*the beginning God created the Heauen, and the Earth.

2 And the earth was without forme, and voyd, and darkeneſſe was vpon the face of the deepe: and the Spirit of God mooued vpon the face of the waters.

Fig. 1.5 Document page: a sample showing a difficulty in OCR due to graphics [29–32]

documents, Coustaty [29] mentioned in his PhD report (Navidomass project (ANR-06-MDCA-012)) that images of these documents are used to identify the history of books. The thesis was dedicated particularly to graphical images i.e., drop-caps appeared mostly as the first character of the paragraph. It has been taken as one of the complex images, since it is composed of different layers of data (images composed of strokes). To extract distinct features from such graphics, unlike the way text recognition has been done, their primary idea is to describe graphics by taking two different layers of information: shapes and lines. Introducing two different bags: bag of patterns and bag of strokes has enriched the quality of the image description. In this project, authors [29–32] have clearly identified the appropriate use of graphics in text recognition (ancient document images).

Besides, especially for historical maps, needless to mention that text recognition from maps is not trivial. This is primarily because of map labels often overlap with other map features, such as road lines (intersections), do not follow a fixed orientation within a map, and can be stenciled and handwritten text [33–38]. Also, many historical scanned maps suffer from poor graphical quality due to bleaching of the original

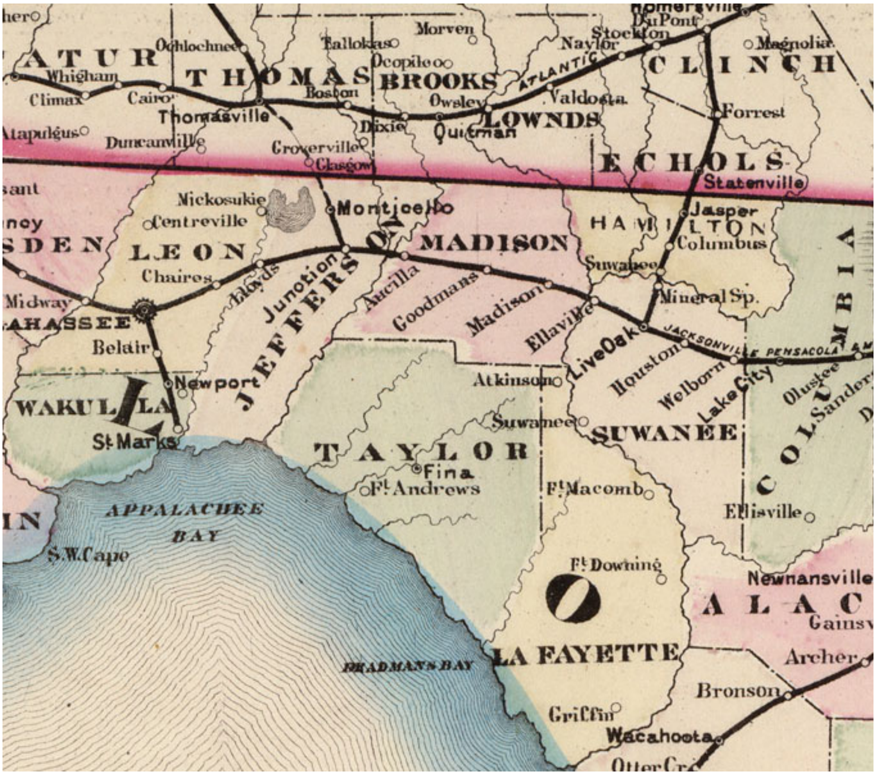


Fig. 1.6 Document page: a sample showing a typical problem of text and graphics recognition from maps [33–36]

paper maps and archiving practices. This presents the idea of text string separation from mixed text/graphics images [7]. In addition, the importance of line structure extraction (line-drawing) was highlighted [39]. Later, text/graphics separation has been revisited in the literature [40] (Fig. 1.6).

Like in text processing, in both articles [1, 8], for graphics processing, basic tasks are image segmentation, layout understanding and graphics recognition. Once again, not surprisingly, as mentioned earlier, input images are scanned paper documents. More detailed information can be found in Chap. 2.

1.3 Summary

Text processing cannot just go alone, and it holds same for graphics processing [45]. In this chapter, we have learned that text and graphics processing complement to each other. In other words, even though a large portion of the document image contains textual information, we find that document image analysis is not just limited to text processing. As an example, graphics processing can help rich the optical character recognition quality. At the same time, it is important to note that text/graphics separation do not really make a difference with text/graphics recognition. Use of the term ‘separation’ and ‘recognition’ vary from an application to another. Besides, document layout understanding eases further issues in both cases: text and graphics processing. It is important to note that this chapter has taken care of several different topics even though they do not typically fall under the graphics recognition framework (Figs. 1.7 and 1.8).

Next chapter will be focussed on graphics recognition, retrieval and spotting, in reference to what we have discussed in this chapter. Learning graphical symbols and spotting them in the scanned document, such as architectural floor plan and mechanical drawing (line drawings, for instance) could be interesting, regardless of their versions: handwritten or machine-printed [46]. Interpreting such a document image can be done by recognizing/spotting the graphical symbols, such as door, bath-tub, stove, sink and walk-in closet [41, 47]. Recently, an end-to-end procedural floor plan design has been addressed, where a wide variety of input image styles and building shapes, including non-convex polygons are handled for architectural tools and digital content generation [48, 49]. Arrowhead can be taken as an important entity that can be used to interpret mechanical drawings (engineering) [43, 44, 50], in addition to other graphical symbols and/or visual cues/primitives, such as line, arc and circle. Analyzing electrical circuit diagram could be another useful application in the graphics recognition framework [46, 47, 51–59].

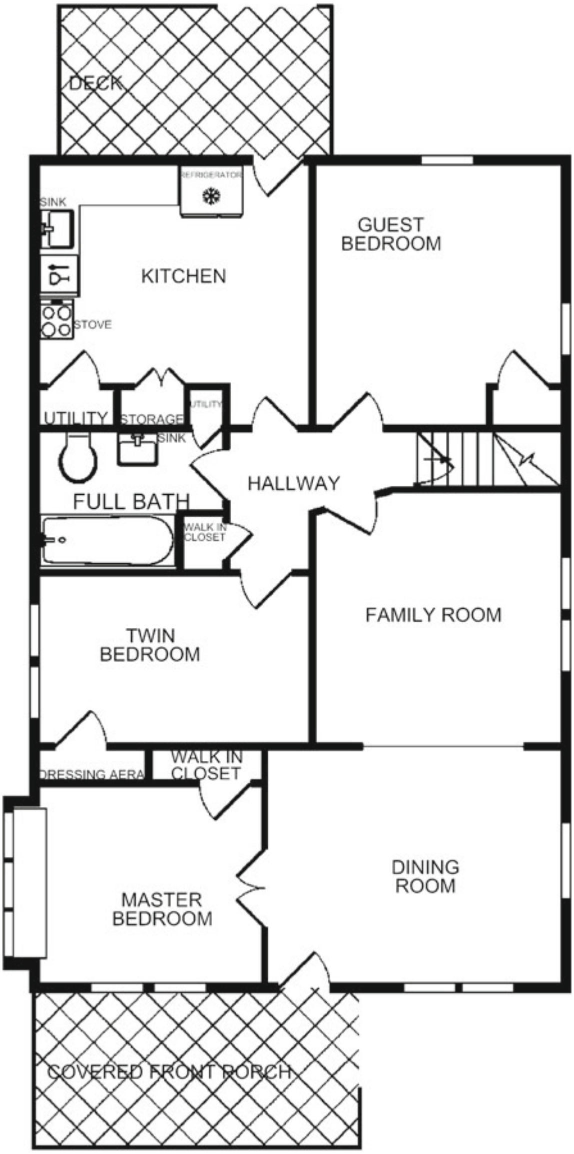


Fig. 1.7 Document page: a sample showing a typical problem of graphics (graphical symbol) recognition and/or spotting from the architectural floor plan [41]

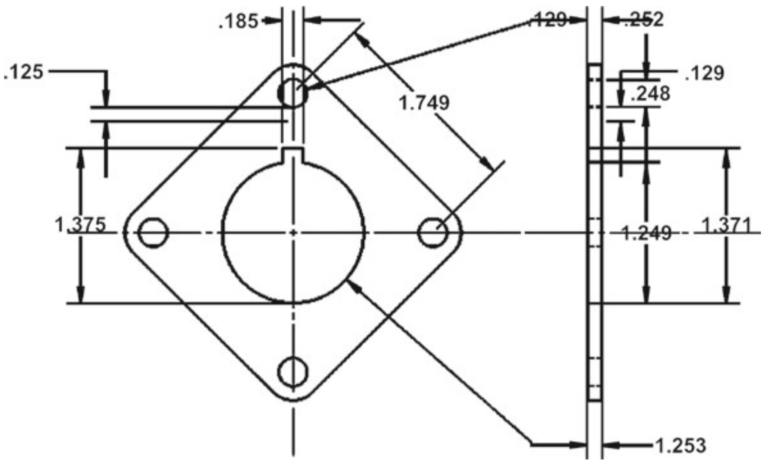


Fig. 1.8 Document page (mechanical drawing) [41]: a sample showing a typical problem of graphics processing. Beside text processing, arrowhead can be taken as an entity to interpret mechanical drawings (engineering) [42–44]. Note that information (data) extraction based on what has been pointed by the arrowhead are important

References

1. R. Kasturi, L. O'Gorman, V. Govindaraju, Document image analysis: a primer. Character Recognit. **27**(1), 3–22 (2002)
2. B. Klein, S. Agne, A. Dengel, Results of a study on invoice-reading systems in Germany, in *Simone Marinai and Andreas Dengel. Proceedings of International Workshop on Document Analysis Systems*. Lecture Notes in Computer Science, vol. 3163 (Springer, Berlin, 2004), pp. 451–462
3. K.C. Santosh, A. Belaïd, Document information extraction and its evaluation based on client's relevance, in *12th International Conference on Document Analysis and Recognition* (2013), pp. 35–39
4. K.C. Santosh, A. Belaïd, Client-driven content extraction associated with table, in *Proceedings of the 13th IAPR International Conference on Machine Vision Applications* (2013), pp. 277–280
5. K.C. Santosh, A. Belaïd, Pattern-based approach to table extraction, in *Pattern Recognition and Image Analysis - 6th Iberian Conference, IbPRIA 2013, Funchal, Madeira, Portugal, June 5–7, 2013. Proceedings* (2013), pp. 766–773
6. K.C. Santosh, g-DICE: graph mining-based document information content exploitation. Int. J. Doc. Anal. Recognit. (IJDAR) **18**(4), 337–355 (2015)
7. L.A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans. Pattern Anal. Mach. Intell. **10**(6), 910–918 (1988)
8. G. Nagy, Twenty years of document image analysis in PAMI. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 38–62 (2000)
9. A. Dengel, G. Barth, Anastasil: hybrid knowledge-based system for document layout analysis, in *Proceedings of the 11th International Joint Conference on Artificial Intelligence (IJCAI'89)*, vol. 2 (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1989), pp. 1249–1254
10. H.S. Baird, Anatomy of a versatile page reader. Proc. IEEE **80**(7), 1059–1065 (1992)
11. L. O'Gorman, The document spectrum for page layout analysis. IEEE Trans. Pattern Anal. Mach. Intell. **15**(11), 1162–1173 (1993)

12. S.-W. Lee, D.-S. Ryu, Parameter-free geometric document layout analysis. IEEE Trans. Pattern Anal. Mach. Intell. **23**(11), 1240–1256 (2001)
13. F. Shafait, Geometric layout analysis of scanned documents. PhD thesis, Kaiserslautern University of Technology, Germany (2008)
14. A.L. Spitz, Correcting for variable skew, in *Proceedings of the 5th IAPR International Workshop on Document Analysis Systems, Princeton, NJ (USA)*, ed. by D. Lopresti, J. Hu, R. Kashi. Lecture Notes in Computer Science, vol. 2423 (Springer, Berlin, 2002), pp. 179–187
15. A. Belaïd, K.C. Santosh, V. Poulain D'Andecy, Handwritten and printed text separation in real document, in *Proceedings of the 13th IAPR International Conference on Machine Vision Applications, MVA 2013, Kyoto, Japan, May 20–23, 2013* (2013), pp. 218–221
16. X. Peng, S. Setlur, V. Govindaraju, R. Sitaram, Handwritten text separation from annotated machine printed documents using Markov Random Fields. Int. J. Doc. Anal. Recognit. (IJDAR) **16**(1), 1–16 (2013)
17. A. Alaei, M. Delalandre, A complete logo detection/recognition system for document images, in *2014 11th IAPR International Workshop on Document Analysis Systems* (2014), pp. 324–328
18. R. Jain, D. Doermann, Logo retrieval in document images, in *2012 10th IAPR International Workshop on Document Analysis Systems* (2012), pp. 135–139
19. A. Alaei, P.P. Roy, U. Pal, Logo and seal based administrative document image retrieval: a survey. Comput. Sci. Rev. **22**, 47–63 (2016)
20. K. Ubul, G. Tursun, A. Aysa, D. Impedovo, G. Pirlo, T. Yibulayin, Script identification of multi-script documents: a survey. IEEE Access **5**, 6546–6559 (2017)
21. Sk Md Obaidullah, C. Halder, K.C. Santosh, N. Das, K. Roy, PHDIndic_11: page-level handwritten document image dataset of 11 official indic scripts for script identification. Multimedia Tools Appl. **77**(2), 1643–1678 (2018)
22. Sk Md Obaidullah, K.C. Santosh, C. Halder, N. Das, K. Roy, Automatic indic script identification from handwritten documents: page, block, line and word-level approach. Int. J. Mach. Learn. Cybern. (2017)
23. F. Shafait, J. van Beusekom, D. Keysers, T.M. Breuel, Document cleanup using page frame detection. Int. J. Doc. Anal. Recognit. (IJDAR) **11**(2), 81–96 (2008)
24. F. Shafait, T.M. Breuel, The effect of border noise on the performance of projection-based page segmentation methods. IEEE Trans. Pattern Anal. Mach. Intell. **33**(4), 846–851 (2011)
25. M. Agrawal, D. Doermann, Clutter noise removal in binary document images. Int. J. Doc. Anal. Recognit. (IJDAR) **16**(4), 351–369 (2013)
26. R. Martens, L. Claesen, Incorporating local consistency information into the online signature verification process. Int. J. Doc. Anal. Recognit. **1**(2), 110–115 (1998)
27. R. Jayadevan, S.R. Kolhe, P.M. Patil, U. Pal, Automatic processing of handwritten bank cheque images: a survey. Int. J. Doc. Anal. Recognit. (IJDAR) **15**(4), 267–296 (2012)
28. D. Rivard, E. Granger, R. Sabourin, Multi-feature extraction and selection in writer-independent off-line signature verification. Int. J. Doc. Anal. Recognit. (IJDAR) **16**(1), 83–103 (2013)
29. M. Coustaty, Contribution à l'analyse complexe de documents anciens, application aux lettrines. (Complex analysis of historical documents, application to lettrines). PhD thesis, University of La Rochelle, France (2011)
30. M. Coustaty, R. Pareti, N. Vincent, J.-M. Ogier, Towards historical document indexing: extraction of drop cap letters. IJDAR **14**(3), 243–254 (2011)
31. M. Coustaty, K. Bertet, M. Visani, J.-M. Ogier, A new adaptive structural signature for symbol recognition by using a Galois lattice as a classifier. IEEE Trans. Syst. Man Cybern. Part B **41**(4), 1136–1148 (2011)
32. M. Clément, M. Coustaty, C. Kurtz, L. Wendling, Local enlacement histograms for historical drop caps style recognition, in *14th IAPR International Conference on Document Analysis and Recognition* (2017), pp. 299–304
33. Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques. ACM Comput. Surv. **47**(1), 1:1–1:44 (2014)
34. Y.-Y. Chiang, S. Leyk, N.H. Nazari, S. Moghaddam, T.X. Tan, Assessing the impact of graphical quality on automatic text recognition in digital maps. Comput. Geosci. **93**(C), 21–35 (2016)

35. Y.-Y. Chiang, C.A. Knoblock, Recognizing text in raster maps. Geoinformatica **19**(1), 1–27 (2015)
36. J.H. Uhl, Extracting human settlement footprint from historical topographic map series using context-based machine learning. IET Conf. Proc. (2017), pp. 15(6 .)–15 (6 .)(1)
37. Y.-Y. Chiang, Unlocking textual content from historical maps - potentials and applications, trends, and outlooks, in *Recent Trends in Image Processing and Pattern Recognition*, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi (Singapore, 2017), pp. 111–124
38. G. Nagy, A. Samal, S. Seth, T. Fisher, E. Guthmann, K. Kalafala, L. Li, S. Sivasubramaniam, Y. Xu, Reading street names from maps - technical challenges, in *GIS/LIS* (1997)
39. T. Kaneko, Line structure extraction from line-drawing images. Pattern Recognit. **25**(9), 963–973 (1992)
40. K. Tombre, S. Tabbone, L. Pélissier, B. Lamiroy, Ph. Dosch, Text/graphics separation revisited, in *Proceedings of the 5th IAPR International Workshop on Document Analysis Systems, Princeton, NJ (USA)*, ed. by D. Lopresti, J. Hu, R. Kashi. Lecture Notes in Computer Science, vol. 2423 (Springer, Berlin, 2002), pp. 200–211
41. M. Delalandre, E. Valveny, T. Pridmore, D. Karatzas, Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems. Int. J. Doc. Anal. Recognit. **13**(3), 187–207 (2010)
42. W. Min, Z. Tang, L. Tang, Recognition of dimensions in engineering drawings based on arrowhead-match, in *Proceedings of 2nd International Conference on Document Analysis and Recognition, Tsukuba (Japan)* (1993), pp. 373–376
43. L. Wendling, S. Tabbone, Recognition of arrows in line drawings based on the aggregation of geometric criteria using the Choquet integral, in *Proceedings of 7th International Conference on Document Analysis and Recognition, Edinburgh (Scotland, UK)* (2003), pp. 299–303
44. L. Wendling, S. Tabbone, A new way to detect arrows in line drawings. IEEE Trans. Pattern Anal. Mach. Intell. **26**(7), 935–941 (2004)
45. B.B. Chaudhuri, *Digital Document Processing: Major Directions and Recent Advances (Advances in Pattern Recognition)* (Springer, New York, 2006)
46. M. Rusiñol, J. Lladós, *Symbol Spotting in Digital Libraries: Focused Retrieval over Graphic-rich Document Collections* (Springer, London, 2010)
47. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: Bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. and Artif. Intell. **28**(06), 1450017 (2014)
48. L.-P. de las Heras, S. Ahmed, M. Liwicki, E. Valveny, G. Sánchez, Statistical segmentation and structural recognition for floor plan interpretation. Int. J. Doc. Anal. Recognit. (IJDAR) **17**(3), 221–237 (2014)
49. D. Camozzato, L. Dihl, I. Silveira, F. Marson, S.R. Musse, Procedural floor plan generation from building sketches. Vis. Comput. **31**(6–8), 753–763 (2015)
50. G. Priestnall, R.E. Marston, D.G. Elliman, Arrowhead recognition during automated data capture. Pattern Recognit. Lett. **17**(3), 277–286 (1996)
51. K.C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2009), pp. 1330–1334
52. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis). PhD thesis, University of Lorraine, France (2011)
53. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2011), pp. 228–237
54. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
55. K.C. Santosh, L. Wendling, B. Lamiroy, Relation bag-of-features for symbol retrieval, in *12th International Conference on Document Analysis and Recognition* (2013), pp. 768–772
56. K.C. Santosh, B. Lamiroy, L. Wendling, DTW-radon-based shape descriptor for pattern recognition. Int. J. Pattern Recognit. Artificial Intell. **27**(3), 1350008 (2013)

57. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, New York, 2015), pp. 1–22
58. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi Communications in Computer and Information. Science **709**, 3–15 (2017)
59. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)

Chapter 2
Graphics Recognition

2.1 Graphical Symbols

Visual cues and/or designs that are interpreting information about specific contexts refer to graphical symbols. In general, they are two-dimensional shapes (in terms of geometry) in addition to their composition in the highest contextual level of information. It is required to have automatic graphics interpretation and recognition as it happens to be in a variety of applications, such as

(a) engineering drawings and architectural drawings [1–7],
(b) electrical circuit diagrams [8–17],
(c) line drawings [18–21],
(d) musical notations [22, 23],
(e) maps (historical) and road signs [24–30],
(f) mathematical expressions [31],
(g) logos [32–34], and
(h) optical characters that are rich in graphics [35–40].

This book will not consider all topics (mentioned above) even though they fall under the graphics recognition framework. The book will be more focused on those graphical symbols used in electrical circuit diagrams, engineering and architectural drawings, and line drawings regardless of their versions: handwritten or machine-printed.

Following Chap. 1, graphics recognition has been one of the intensive research topics since the 70s in the pattern recognition (PR) and document image analysis (DIA) community [41–44]. In 1998, the following statement: “*none of these methods works in general*” influenced researches: what we have done so far and what/where we have/are now? [45, 46]. The statement helped move further [46, 50]. Further, the usefulness of graphics recognition has been reported in the year 2015 [50] and survey has been made in the same year [16].

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_2

2.2 Basics to Graphics Recognition

Not a surprise, graphics are combined with texts in addition to colors. This means that graphics provide more information, i.e., a picture speaks thousands of words. If we do not consider a few generic techniques that are under the DIA framework, text recognition can be taken as different side of the DIA work with respect to graphical symbol recognition. However, their boundary is not straightforward or separable. More often, researchers observed that their solutions complement each other [41, 44, 51]. Therefore, needless to mention, text analysis in graphics requires special attention [35]. To understand the importance of graphics recognition, one should be able to understand that the graphical symbol recognition (or any meaningful shapes/parts/regions) has been the subject of several different projects (as mentioned in Sect. 2.1) [2, 51–56]. Generally speaking, these proposed approaches are roughly categorized into the following:

(a) data acquisition,
(b) data preprocessing, and
(c) data representation/description and recognition/classification.

The first two items: data acquisition and preprocessing techniques—which can be considered as a unit, in a broad sense—are application dependent. In some cases, where data are clean, preprocessing may not be required. Text/graphics separation refers to document image segmentation [57]; and they basically decompose document image into two layers so that one can consider the layer, where graphics lie. More detailed study on text/graphics separation can be found in [58]. In the framework of data description, graphical symbols are described either in terms of a set of numbers, i.e., feature vector by taking into account the overall shape (statistical data representation) or in terms of structured forms (graph representation) by taking visual cues/words that compose whole graphical symbol. Besides, the rule-based representation can describe the overall shape of the pattern. In both cases, visual cues/words are found to be application dependent. In the decision process, matching techniques often follow the way how graphical symbols are represented. In general, data description (or representation) is said to be good if it can maximize the interclass distance and minimize the intraclass distance [47]. The term good refers to how compact the feature vector is and how well two feature vectors can be discriminant. Existing approaches, broadly speaking, can be divided into three different categories: (i) statistical, (ii) structural, and (iii) syntactic. These categories are assumed to be based on feature-based matching concept. Before proceeding to upcoming chapters, it is found that neither of the techniques alone can help achieve expected performance. This means that, in the literature, we have observed a common trend, where authors combine different techniques from different categories: statistical, structural, and syntactic. Integrating/combining them (statistical and structural, for instance) aims at taking advantage of both techniques [11, 12, 15–17]. Meaning, it is worth to integrate if they compliment each other and satisfy the utility functions that can reach the goal. More detailed information can be found in Chap. 3.

2.3 Contests and Real-World Challenges in Graphics Recognition

In Chap. 1, an importance of graphics processing has been outlined in the framework DIA. Considering the same, this section aims to include graphics recognition contests and check whether they have been addressing real-world projects. Since 1995, the international association of pattern recognition (IAPR) sponsored graphics recognition (GREC) workshops, supported by technical committee 10 (TC-10: http://iapr-tc10.univ-lr.fr/) organized several contests in the framework of graphics recognition. The contests are not limited to graphical symbol recognition, retrieval, and spotting; they also came up with several other contests, such as arc and line segmentations.

While considering all contests, the observation can be summarized as follows. In brief, the primary objectives of the GREC contests are to evaluate the state of the art of graphics recognition techniques (plus other related works), to generate performance evaluation tools, techniques, and to provide datasets for future extensions [5, 59–61]. The contests do not just provide summary of results from the participated institutions/researchers but also provide datasets and guide for evaluating their tools, i.e., a comprehensive protocol.

In the following, the list of contests can be enumerated as follows:

(a) *GREC'13: Arc and line segmentation contest* [64]
Since geometric primitives, such as line and arc (see Fig. 2.1) helps in automatic conversion of line drawing document images into electronic form, their recognition and/or detection is important. As mentioned in the title, two challenges were proposed: arc segmentation and line segmentation. For these contests, engineering drawings (for arc segmentation challenge) and cadastral maps (for line segmentation challenge) were used. The reported highest possible segmentation accuracies were 54.10 and 66% for arc and line, respectively.

(b) *GREC'11: Arc segmentation contest: performance evaluation on multi-resolution scanned documents* [65]
The sixth edition of the arc segmentation contest was to work on document images with different scanning resolutions. In this contest, altogether nine document images were scanned with three resolutions each and the ground truth images were provided (annotated by the experts). It was observed that the tool that has vectorization techniques/algorithms produced better results on scanned images even with low resolution.

(c) *GREC'11: Symbol recognition and spotting contest* [66]
This contest followed the series started since the GREC'03 workshop (see item

J, below). Unlike the previous ones, it also included symbol spotting problem in addition to the isolated symbol recognition.

(d) *GRECC'09: Arc segmentation contest: performance evaluation on old documents* [67]
This was focused on empirical performance evaluation of raster-to-vector algorithms in the area of graphics recognition. For the contest, old document images were used, where a few commercial software were participated. This helped us check whether automatic vectorization methods (prototypes) reached the maturity as if they could be taken as a commercial software.

(e) *GREC'07: Third contest on symbol recognition* [68]
This contest followed the series started since the GREC'03 workshop (see item J, below). The main different between two contests is changes in test data.

(f) *GREC'07: Arc segmentation contest* [69]
As expected, the idea was to check/compare different state-of-the-art systems: arc segmentation. Four algorithms were tested.

(g) *GREC'05: Arc segmentation contest* DBLP:conf/grec/Wenyin05
In the sixth series of graphics recognition workshop organized by IAPR TC10, this was the third arc segmentation contest, where three tools were participated. In addition, second evaluation of the RANVEC and the arc segmentation contest was reported [70]. In the latter case, important facts are recalled and provided detailed information about changes made on the system compared to GREC‘01.

(h) *GREC'05: Symbol recognition contest* [71]
This was the second symbol recognition contest, and organizers brought general principles of both contests: GREC'03 and GREC'05.

(i) *GREC'03: Arc segmentation contest* [72]
In the fifth series of graphics recognition workshop organized by IAPR TC10, the arc segmentation contest provided rules, performance metrics and data.

(j) *GREC'03: Symbol recognition contest* [63]
This was the first international symbol recognition contest, where organizers described the framework of the contest: goals, symbol types and evaluation protocol. As mentioned in their report, the idea was to make participants ready for the upcoming contest. Organizers provided the way they have built the database and the methods they used to add noise. This helped researchers evaluate the robustness of their methods/algorithms.

(k) *GREC'01: Arc segmentation contest* [73–75]
As the fourth in the series of graphics recognition contests organized by IAPR TC10, the first arc segmentation contest was held in association with the GREC'01 workshop. In addition to general rules, organizers provided arcs and circles in engineering drawings and other scanned images containing line-work for the test. We find that the tool that has an algorithm to vectorize binary images smooths the vectors to a sequence of small straight-like lines received better results. We note that engineering drawings were mostly used.

(l) *GREC'97: International graphics recognition contest—raster-to-vector conversion* [76, 77]
It is important to note that vectorization techniques can help boost the perfor-

mance of the further processes, such as arc segmentation. Based on the experience, GREC team started with the idea of raster-to-vector conversion in the second series of graphics recognition workshop.
Further, they have defined a computational protocol to evaluate performance for systems that convert raster data to vector. In this contest, continuous and dashed lines, arcs, and circles and text regions were considered as the graphical entities.

(m) *GREC'95: Dashed line detection* [78–80]
The first graphics recognition contest was dashed line detection, where test image generator created random line patterns with a few constraints.
At this point, it is important to note that visual cues, such as dashed line, are essential for high-level technical drawing understanding if we are able to detect/segment them. The idea was to automatically segment them since machine vision is required for a large amount of data. As a consequence, the contest was about automatic detection of dashed lines on test drawings at three difficulty levels: simple, medium, and complex. They basically have dashed and dash-dotted lines in straight and curved shapes, including interwoven texts.

In the year 2007 (GREC'07), Prof. Tombre highlighted an important issue that whether graphics recognition is an unidentified scientific object [81]. In this discussion, he has clearly mentioned the fact as follows. Since the day when Prof. Kasturi gave a new start to a technical committee of the IAPR, namely, TC10 on line drawing interpretation, researchers have focused on graphics-rich documents and more specific issues, such as raster-to-graphics conversion, text/graphics separation and symbol recognition/localization. To emphasize new focus, TC10 was titled as the technical committee of graphics recognition. Meaning, GREC started since then with a series of LNCS volumes.[1] No doubt that graphics recognition contests provide a clear benchmark for researchers and help proceed in reference to what has been done in the past.

Researchers do not really see any doubt on the growing interest/importance of the field: graphics recognition. A few specialized areas, such as telephone and power companies that hold huge numbers of drawings with the same syntax/format and/or appearance are interesting applications. Automatic data conversion helps develop processing tool cost-effective, since these data are rich graphics and graphical symbol as a query is possible. In other words, it is required to convert paper documents that contain graphics into electronic formats, which is becoming more and more useful in a variety of applications.

Besides, in recent years, we have observed the significance of "end-to-end document analysis benchmarking" and "open resource sharing repository" to advance as well as to facilitate fair comparison [82, 83]. More information can be gathered from the project called "Document Analysis and Exploitation" (DAE).[2]

Back to the real-world problems, symbol recognition is not straightforward as shown in Fig. 2.6. In general, common problems are recognition and localization (more often, we call it spotting) of graphical symbols in electronic documents, in

[1] URL: http://dblp.uni-trier.de/db/conf/grec/index.html.

[2] URL: http://dae.cse.lehigh.edu/DAE/.

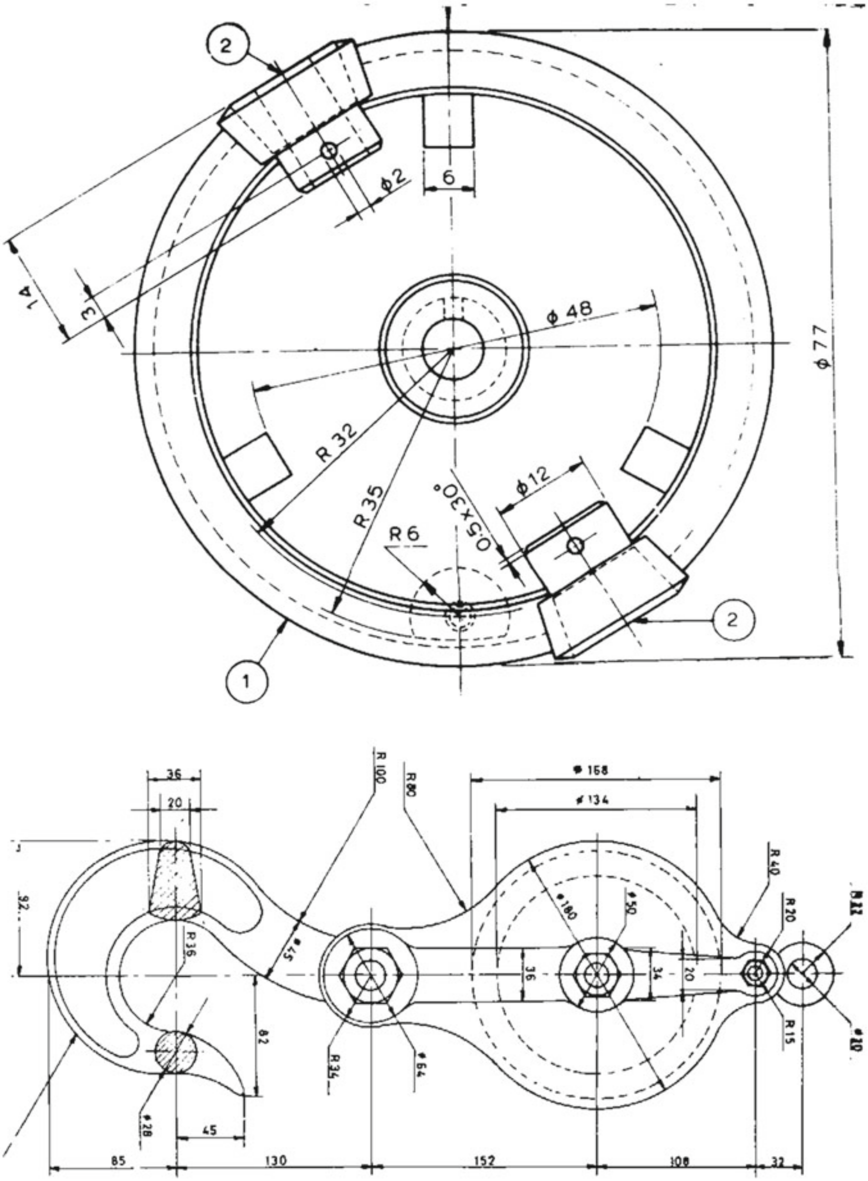


Fig. 2.1 A few test images from GREC'11: arc segmentation contest [65]

architectural floor plans (see Figs. 2.2 and 2.3), wiring diagrams and network drawings (see Figs. 2.4 and 2.5) [5, 12, 47, 66].

Beside the lineal and fully isolated graphical symbol recognition (see Fig. 2.6), in this book, a new challenging problem will be highlighted (see Fig. 2.7), where the

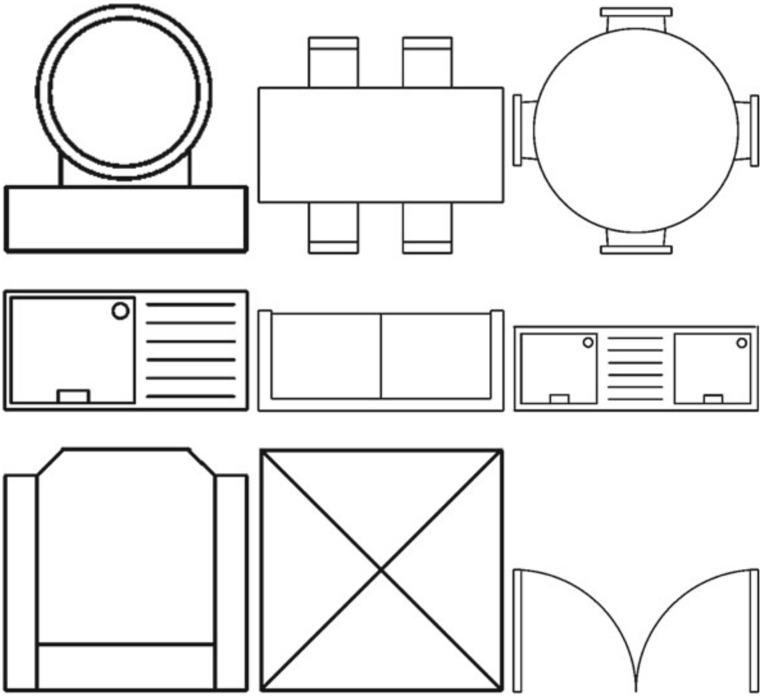


Fig. 2.2 A few test images from GREC'11: symbol segmentation contest [66]

dataset is composed of a variety of symbols, such as linear (fully isolated), complex, and composite (with texts in it). Note that the characteristics of the problem are not different than what have been addressed in a series of graphics recognition contests/workshops. Primarily, the difference lies in the dataset. These samples (called by the name FRESH dataset) are taken from the book [84]. Two different symbols from different classes look very similar in shape (with slight changes) [12, 85–87]. Graphical cues and/or texts can also be present. They do not always connect with the graphical symbols we are looking for; they can also be isolated in the same image. For such a case, an isolated graphical symbol (or known part of it) can be applied for two different reasons: (i) to recognize similar symbols; and (ii) to detect known and meaningful parts/regions [17]. Detecting meaningful parts/regions with respect to the applied query symbol refers to symbol spotting. Therefore, not to be confused, we are not just limited to symbol recognition problem. We are also required to spot the meaningful parts/regions that can convey contextual information about the graphical documents. Further, it is always interesting to check the similarity between two different symbols that are taken from different contexts. The latter issue is taken as one of the open challenging issues in the literature. On the whole, the task has been referred to as either the parts/regions or the complete symbol recognition [5, 12, 47, 88–90]. A priori knowledge about graphical symbol can help decide the techniques for data representation and recognition.

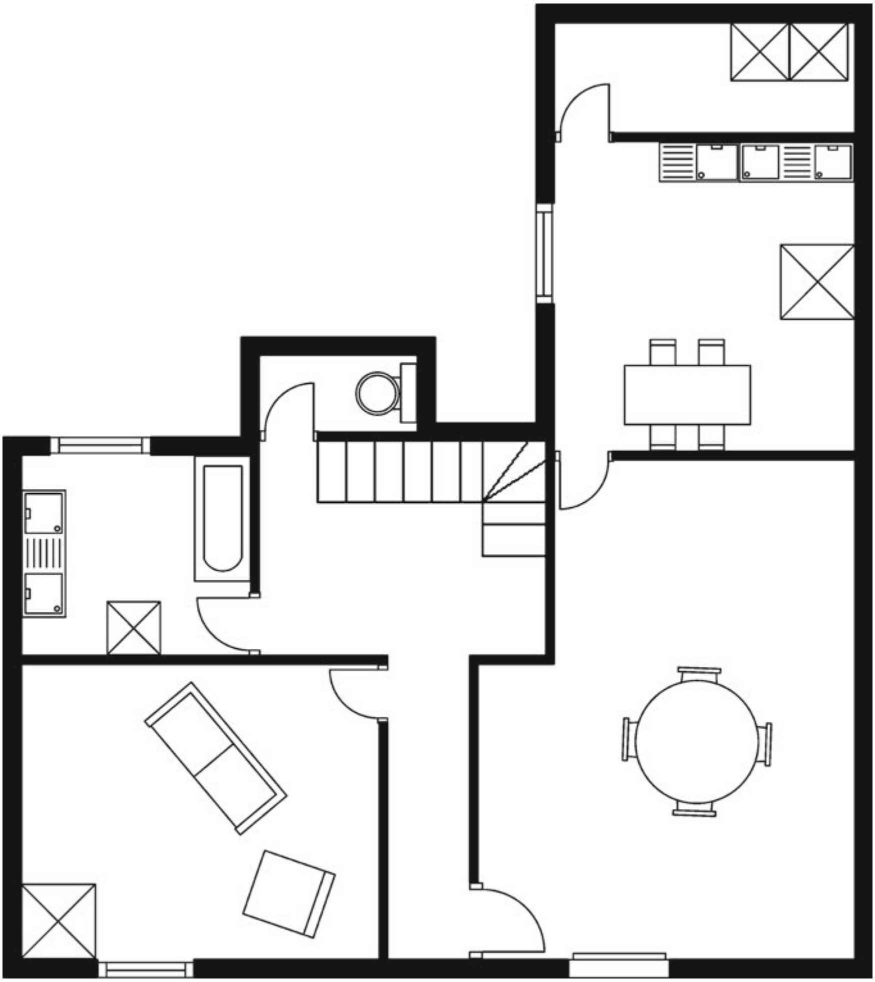


Fig. 2.3 An example graphical symbol spotting/localization in the architectural floor plan [5, 66]

2.4 Graphical Symbol Recognition, Retrieval, and Spotting

Under the scope of pattern recognition, symbol recognition is a particular application, where test input patterns are classified as one of many classes that are predefined symbol types (ground truths) in the particular application domain. Graphical symbols do not necessarily be a complete symbol as shown in Figs. 2.2 and 2.4. It can be other visual cues or visual primitives, such as arc, lines, and circle that can be used to interpret complete document images. In a broad sense, in reference to [88], symbols can be defined as the graphical entities which hold a semantic meaning in any specific domain, where logos, silhouettes, musical notes, and simple line segment groups with

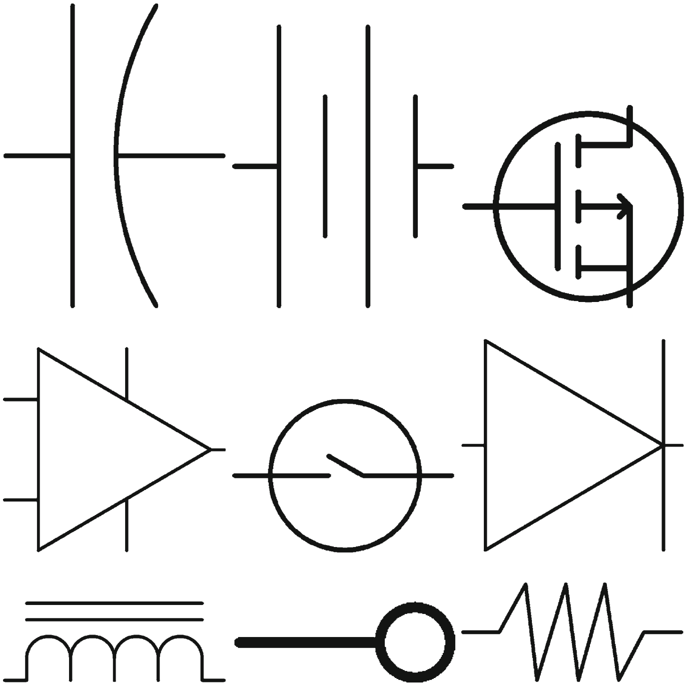


Fig. 2.4 A few test images from GREC'11: symbol segmentation contest (electrical symbols) [66]

an engineering, electronics, or architectural flair constitute are some examples of symbols that have been investigated recently by the graphics recognition community (see previous Sect. 2.1). Extracting/retrieving similar documents, based on visual cues (graphical primitives) can be considered as graphical symbol retrieval. This, of course, requires a clear knowledge of symbol spotting.

In what follows, the brief research standpoints on graphics recognition are summarized. More detailed information can be found in [16, 17].

2.5 Research Stand Points: A Quick Overview

Before we move to Chap. 3, generally speaking, the whole graphical symbol recognition process is based on either

(a) alignment of features between a query and template symbols, i.e., computing distance between two feature vectors; or
(b) comparing decomposed parts, i.e., meaningful visual cues/words, such as lines, arcs, and circles, and the relations (spatial relations) between them.

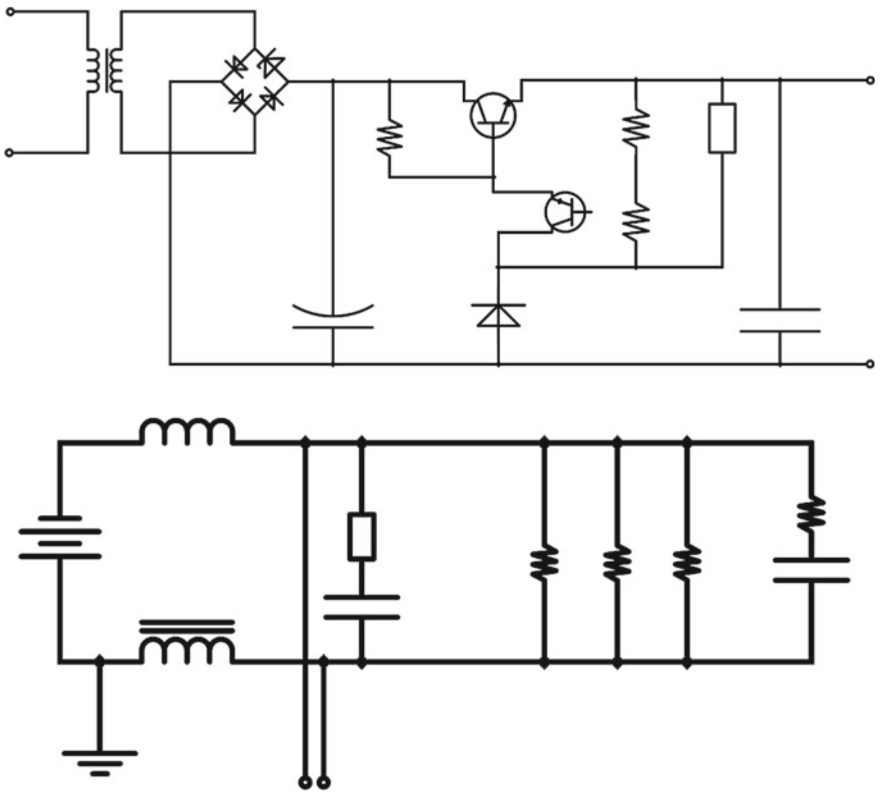


Fig. 2.5 Few test images (electrical circuit diagram): GREC'11: symbol segmentation contest [66]. An interesting problem to see how one can go for symbol spotting/localization

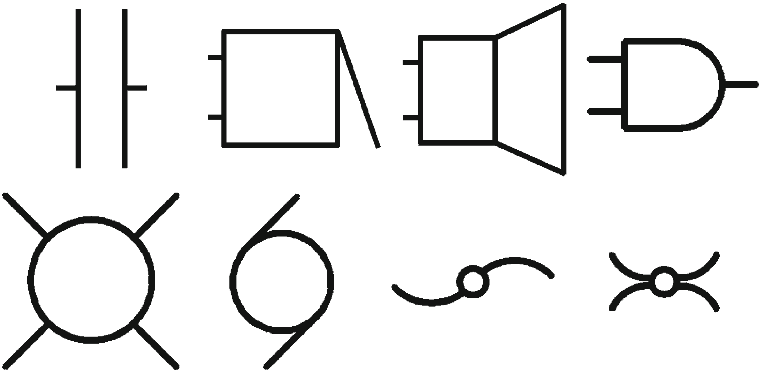
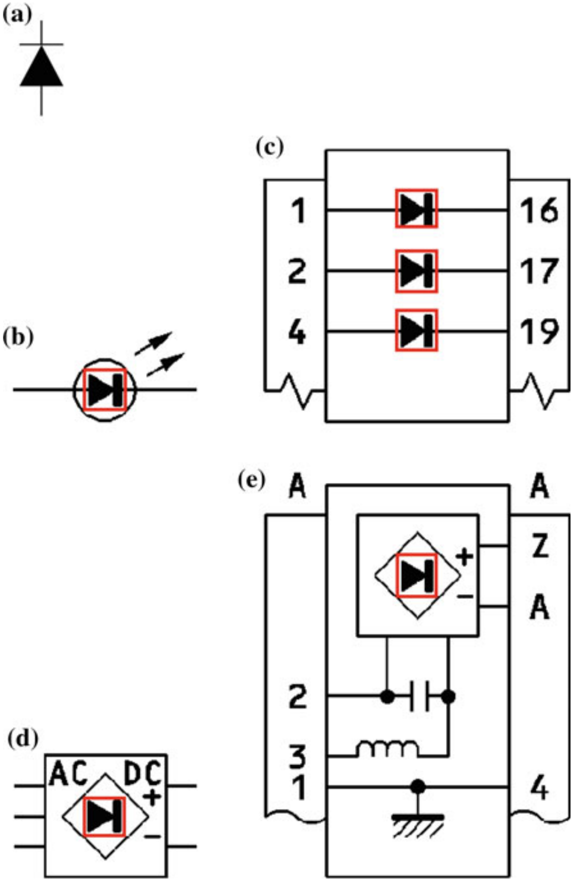


Fig. 2.6 GREC'03: illustrating lineal and fully isolated graphical symbols [62]

Fig. 2.7 An example of **a** a query and **b**–**e** graphical symbol or meaningful parts/regions spotting. Further, it also illustrates the complexity of the dataset [12, 84]. Graphical elements in the red box the detected regions in accordance with what has been applied as a query



These are commonly described within the framework of statistical and structural approaches, respectively. A quick overview can be found in the previous work [17]. In statistical approach, shape descriptors are widely used. A quick overview the most commonly used shape descriptors for graphical symbol recognition is provided in [91]. On the other hand, structural approaches allow low-level primitives or visual cues analysis so that recognizing graphical symbols and/or localizing known visual parts are possible. Not to be confused, ROIs refer to meaningful parts. Like in other domain, the concept is in the scope of regions-of-interest (ROI) analysis and labeling. This means that one can take a graphical symbol as a set of visual cues or meaningful parts, such as arcs, lines, triangles and rectangles [3, 12, 92]. The set also includes higher level visual cues like loops. Their interpretations, however, depend on the dataset and the context. The context can be either local or global. Therefore, visual cues in graphical symbol recognition, on the whole, can be considered as one of the key steps toward document image understanding and content interpretation. Considering both approaches into account, we have observed the use of their best possible combination [12, 15]. For this, a clear statement can be taken from the GREC'10 [24] and a part of it is outlined as follows:

> *... the recurring wish for methods capable of efficiently combining structural and statistical methods'* and '*the very structural and spatial nature of the information we work with makes structural methods quite natural in the community.*

An extension, i.e., symbol spotting is possible, but one can view this as a kind of graphical symbol retrieval problem [5, 14, 88, 93, 94] that is basically user guided. Additionally, using the local descriptors like scale invariant feature transform (SIFT) and other techniques like bag-of-features (BOFs), recognition/retrieval process can be accomplished. In both cases, it is possible to avoid segmentation process, i.e., primitive and/or region extraction. The questions, such as "what technique does how much/well in which context?" has not been well answered yet.

No doubt (see Sect. 2.3), graphics recognition has a rich literature with several different techniques [47, 50, 95, 96]. More often, symbol recognition methods are not generic enough to be used for different purposes and/or datasets. However, these methods not require a large set of parameters, and sometimes, they are parameter-free, i.e., easy to implement. This means that methods are data dependent. Another reason could be the restriction posed by the industrial needs. Industrial projects require automated systems with higher accuracy so that the cost of human intervention can be reduced. This will ensure its effectiveness as well. As a result, graphical symbol recognition techniques might be tuned into process data under several different circumstances. Industrial projects are related to information retrieval and/or document reverse engineering. Such projects require powerful computers (high-performance computing (HPC) machines in addition to huge storage capacity. Within this framework, scientific community provides serious attention in recognizing symbols in

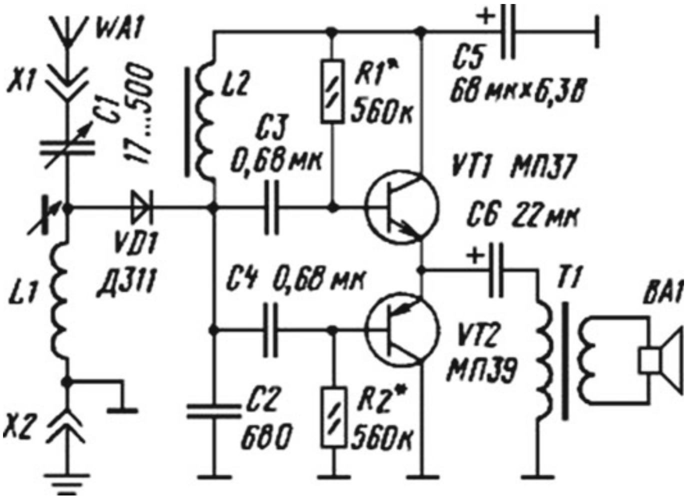


Fig. 2.8 Handwritten electrical circuit diagram

document images [96–99]. Note that the processed images are not necessarily be technical documents. For graphics recognition, it is required to have consistent advances in research so that scalability issue can be addressed. The scalable property can help reach the industrial needs and/or expectations. This also explains why well-known approaches were very specific and were guided by a priori knowledge. A priori knowledge can be either context or the source/complexity of the data. Both of them can be used as well. This will definitely help us move forward to other similar problems, such as digitization of the handwritten electrical circuit diagrams (see Fig. 2.8). Digitizing handwritten electrical circuit diagrams in accordance with the floor plan can help automate the full residence needs (depends on the regional variation, i.e., geography).

2.6 Summary

In this chapter, we have started with the conventional definition of graphical symbols, the location of graphics recognition in DIA and its major processing units, several international contests that are related to graphics recognition and their importance, and a quick overview of research standpoints (from the author's perspective). On the whole, we have discussed the importance of graphics recognition in the DIA framework. Our next chapter will discuss graphics recognition systems and validation/evaluation protocols.

References

1. P.M. Devaux, D.B. Lysak, R. Kasturi, A complete system for the intelligent interpretation of engineering drawings. Int. J. Doc. Anal. Recognit. **2**(2/3), 120–131 (1999)
2. J. Lladós, E. Martí, J.J. Villanueva, Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. IEEE Trans. Pattern Anal. Mach. Intell. **23**(10), 1137–1143 (2001)
3. K. Ph Dosch, C. Tombre, G.Masini Ah-Soon, A complete system for analysis of architectural drawings. Int. J. Doc. Anal. Recognit. **3**(2), 102–116 (2000)
4. E. Valveny, E. Martí, A model for image generation and symbol recognition through the deformation of lineal shapes. Pattern Recognit. Lett. **24**(15), 2857–2867 (2003)
5. M. Delalandre, E. Valveny, T. Pridmore, D. Karatzas, Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems. Int. J. Doc. Anal. Recognit. **13**(3), 187–207 (2010)
6. L.S. de las Heras, S. Ahmed, M. Liwicki, E. Valveny, G. Sánchez, Statistical segmentation and structural recognition for floor plan interpretation. Int. J. Doc. Anal. Recognit. (IJDAR) **17**(3), 221–237 (2014)
7. D. Camozzato, L. Dihl, I. Silveira, F. Marson, S.R. Musse, Procedural floor plan generation from building sketches. Vis. Comput. **31**(6–8), 753–763 (2015)
8. A. Okazaki, T. Kondo, K. Mori, S. Tsunekawa, E. Kawamoto, An automatic circuit diagram reader with loop-structure-based symbol recognition. IEEE Trans. Pattern Anal. Mach. Intell. **10**(3), 331–341 (1988)

9. Guihuan Feng, Christian Viard-Gaudin, Zhengxing Sun, On-line hand-drawn electric circuit diagram recognition using 2d dynamic programming. Pattern Recognit. **42**(12), 3215–3223 (2009)
10. K.C. Santosh, L. Wendling, B. Lamiroy, Using spatial relations for graphical symbol description, in *Proceedings of the IAPR International Conference on Pattern Recognition* (IEEE Computer Society, Washington, 2010), pp. 2041–2044
11. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2011), pp. 228–237
12. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics recognition using spatial relations and shape analysis). PhD thesis, University of Lorraine, France, 2011
13. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
14. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: Bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. Artif. Intell. **28**(06), 1450017 (2014)
15. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
16. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, New York, 2015), pp. 1–22
17. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in *Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers*, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi. Communications in computer and information science, vol. 709 (2017), pp. 3–15
18. W. Min, Z. Tang, and L. Tang, Recognition of dimensions in engineering drawings based on arrowhead-match, in *Proceedings of 2nd International Conference on Document Analysis and Recognition, Tsukuba (Japan)* (1993), pp. 373–376
19. G. Priestnall, R.E. Marston, D.G. Elliman, Arrowhead recognition during automated data capture. Pattern Recognit. Lett. **17**(3), 277–286 (1996)
20. L. Wendling, S. Tabbone, Recognition of arrows in line drawings based on the aggregation of geometric criteria using the Choquet integral, in *Proceedings of 7th International Conference on Document Analysis and Recognition, Edinburgh (Scotland, UK)* (2003), pp. 299–303
21. L. Wendling, S. Tabbone, A new way to detect arrows in line drawings. IEEE Trans. Pattern Anal. Mach. Intell. **26**(7), 935–941 (2004)
22. K. Ng, Music manuscript tracing, in *Graphics Recognition Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon (Springer, Berlin, 2002), pp. 330–342
23. A. Rebelo, G. Capela, J.S. Cardoso, Optical recognition of music symbols: a comparative study. Int. J. Doc. Anal. Recognit. **13**(1), 19–31 (2010)
24. Hanan Samet, Aya Soffer, MARCO: map retrieval by content. IEEE Trans. Pattern Anal. Mach. Intell. **18**(8), 783–798 (1996)
25. Y.-Y. Chiang, S. Leyk, C.A. Knoblock, A survey of digital map processing techniques. *ACM Comput. Surv.*, 47(1), 1:1–1:44 (2014)
26. Y.-Y. Chiang, Unlocking textual content from historical maps - potentials and applications, trends, and outlooks, in *Recent Trends in Image Processing and Pattern Recognition*, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi (Springer, Singapore, 2017), pp. 111–124
27. Y.-Y. Chiang, S. Leyk, N.H. Nazari, S. Moghaddam, T.X. Tan, Assessing the impact of graphical quality on automatic text recognition in digital maps. Comput. Geosci. 93(C), 21–35 (2016)
28. Y.-Y. Chiang, C.A. Knoblock, Recognizing text in raster maps. Geoinformatica **19**(1), 1–27 (2015)
29. J.H. Uhl, Extracting human settlement footprint from historical topographic map series using context-based machine learning, in *IET Conference Proceedings* (2017), pp. 15 (6 .)–15 (6 .)(1)

30. G. Nagy, A. Samal, S. Seth, T. Fisher, E. Guthmann, K. Kalafala, L. Li, S. Sivasubramaniam, Y. Xu, Reading street names from maps - technical challenges, in *GIS/LIS* (1997)
31. B.B. Chaudhuri, Utpal Garain, An approach for recognition and interpretation of mathematical expressions in printed document. Pattern Anal. Appl. **3**(2), 120–131 (2000)
32. A. Alaei, M. Delalandre, A complete logo detection/recognition system for document images, in *2014 11th IAPR International Workshop on Document Analysis Systems* (2014), pp. 324–328
33. R. Jain, D. Doermann. Logo retrieval in document images, in *2012 10th IAPR International Workshop on Document Analysis Systems* (2012), pp. 135–139
34. A. Alaei, P.P. Roy, U. Pal, Logo and seal based administrative document image retrieval: a survey. Comput. Sci. Rev. **22**, 47–63 (2016)
35. M. Coustaty, Contribution à l'analyse complexe de documents anciens, application aux lettrines. (Complex analysis of historical documents, application to lettrines). Ph.D. thesis, University of La Rochelle, France, 2011
36. Mickaël Coustaty, Rudolf Pareti, Nicole Vincent, Jean-Marc Ogier, Towards historical document indexing: extraction of drop cap letters. IJDAR **14**(3), 243–254 (2011)
37. M. Coustaty, K. Bertet, M. Visani, J.-M. Ogier, A new adaptive structural signature for symbol recognition by using a galois lattice as a classifier. IEEE Trans. Syst. Man Cybern. Part B **41**(4), 1136–1148 (2011)
38. M. Clément, M. Coustaty, C. Kurtz, L. Wendling, Local enlacement histograms for historical drop caps style recognition, in *14th IAPR International Conference on Document Analysis and Recognition* (2017), pp. 299–304
39. K.C. Santosh, Character recognition based on dtw-radon, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2011), pp.64–268
40. K.C. Santosh, *Laurent Wendling, Character Recognition Based on Non-linear Multi-projection Profiles Measure* (Science, Frontiers of Computer, 2015), pp. 1–13
41. L.A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans. Pattern Anal. Mach. Intell. **10**(6), 910–918 (1988)
42. G. Nagy, Twenty years of document image analysis in PAMI. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 38–62 (2000)
43. Rangachar Kasturi, Lawrence O'Gorman, Venu Govindaraju, Document image analysis: a primer. Character Recognit. **27**(1), 3–22 (2002)
44. B.B. Chaudhuri, *Digital Document Processing: Major Directions and Recent Advances (Advances in Pattern Recognition)* (Springer, New York, 2006)
45. K. Tombre, Analysis of engineering drawings: state of the art and challenges, in Tombre, Chhabra [100], pp. 257–264
46. K. Tombre, Ten years of research in the analysis of graphics documents: achievements and open problems, in *Proceedings of 10th Portuguese Conference on Pattern Recognition, Lisbon, Portugal* (1998), pp. 11–17
47. J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *Graphics Recognition - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 104–127
48. K. Tombre, Graphics recognition: the last ten years and the next ten years, in *Proceedings of 6th IAPR International Workshop on Graphics Recognition, Hong Kong* (2005), pp. 422–426
24. K. Tombre, Graphics recognition - what else?, in *Graphics Recognition. Achievements, Challenges, and Evolution*, ed. by J.-M. Ogier, W. Liu, J. Lladós, Lecture Notes, in Computer Science, vol. 6020, (Springer, Berlin, 2010), pp. 272–277
50. D. Doermann, K. Tombre, *Handbook of Document Image Processing and Recognition* (Springer, New York Incorporated, 2014)
51. A.K. Chhabra. Graphic symbol recognition: an overview, in *Proceedings of 2nd International Workshop on Graphics Recognition, Nancy (France)* (1997), pp. 244–252
52. D.S. Doermann, An introduction to vectorization and segmentation, in Tombre, Chhabra [100], pp. 1–8

53. R. Kasturi, R. Raman, C. Chennubhotla, L. O'Gorman, Document image analysis: an overview of techniques for graphics recognition, in *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)* (1990), pp. 192–230
54. A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
55. S. Loncaric, A survey of shape analysis techniques. Pattern Recognit. **31**(8), 983–1001 (1998)
56. S. Marshall, Review of shape coding techniques. Image Vis. Comput. **7**(4), 281–294 (1989)
57. T. Taxt, P.J. Flynn, A.K. Jain, Segmentation of document images. IEEE Trans. Pattern Anal. Mach. Intell. **11**(12), 1322–1329 (1989)
58. Salvatore Tabbone, Laurent Wendling, Karl Tombre, Matching of graphical symbols in line-drawing images using angular signature information. Int. J. Doc. Anal. Recognit. **6**(2), 115–125 (2003)
59. M. Delalandre, J.-Y. Ramel, N. Sidere, A semi-automatic groundtruthing framework for performance evaluation of symbol recognition and spotting systems. *Lecture Notes in Computer Science*, vol. 7423 (Springer, Berlin, 2013), pp. 163–172
60. M. Delalandre, E. Valveny, J. Lladós, Performance evaluation of symbol recognition and spotting systems: an overview, in *Proceedings of International Workshop on Document Analysis Systems*, ed. K. Kise, H. Sako (IEEE Computer Society, 2008), pp. 497–505
61. Marçal Rusiñol, Josep Lladós, A performance evaluation protocol for symbol spotting systems in terms of recognition and location indices. Int. J. Doc. Anal. Recognit. **12**(2), 83–96 (2009)
62. GREC. International Symbol Recognition Contest at GREC 2003 (2003)
63. E. Valveny, Ph Dosch, Symbol recognition contest: a synthesis, in *Graphics Recognition: Recent Advances and Perspectives - Selected papers from GREC'03*, ed. by J. Lladós, Y.B. Kwon, Lecture Notes, in Computer Science, vol. 3088, (Springer, Berlin, 2004), pp. 368–385
64. S.S. Bukhari, H.S.M. Al-Khaffaf, F. Shafait, M.A. Osman, A.Z. Talib, Final report of grec'13 arc and line segmentation contest, *Graphics Recognition. Current Trends and Challenges - 10th International Workshop, GREC 2013, Revised Selected Papers* (2013), pp. 234–239
65. H.S.M. Al-Khaffaf, A.Z. Talib, M.A. Osman, Final report of grec'11 arc segmentation contest: Performance evaluation on multi-resolution scanned documents, *Graphics Recognition. New Trends and Challenges - 9th International Workshop, GREC 2011, Revised Selected Papers* (2011), pp. 187–197
66. E. Valveny, M. Delalandre, R. Raveaux, B. Lamiroy, Report on the symbol recognition and spotting contest, *Graphics Recognition. New Trends and Challenges - 9th International Workshop, Revised Selected Papers* (2011), pp. 198–207
67. H.S.M. Al-Khaffaf, A.Z. Talib, M.A. Osman, P.L. Wong, Grec'09 arc segmentation contest: performance evaluation on old documents, *Graphics Recognition. Achievements, Challenges, and Evolution, 8th International Workshop, Selected Papers* (2009), pp. 251–259
68. E. Valveny, P. Dosch, A. Fornés, S. Escalera, Report on the third contest on symbol recognition, *Graphics Recognition. Recent Advances and New Opportunities, 7th International Workshop, Selected Papers* (2007), pp. 321–328
69. F. Shafait, D. Keysers, T.M. Breuel, GREC, arc segmentation contest: evaluation of four participating algorithms. Graphics Recognition. Recent Advances and New Opportunities, 7th International Workshop. Selected Papers **2007**, 310–320 (2007)
70. X. Hilaire, RANVEC and the arc segmentation contest: second evaluation, Graphics Recognition. Ten Years Review and Future Perspectives, 6th Internation Workshop, GREC, Hong Kong, China, August 25–26, 2005. Revised Selected Papers **2005**, 362–368 (2005)
71. P. Dosch, E. Valveny, Report on the second symbol recognition contest, *Graphics Recognition. Ten Years Review and Future Perspectives, 6th Internation Workshop, GREC 2005, Revised Selected Papers* (2005), pp. 381–397
72. L. Wenyin, Report of the arc segmentation contest, in *Graphics Recognition, Recent Advances and Perspectives*, ed. by J. Lladós, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 3088, (Springer, Berlin, 2004), pp. 364–367
73. L. Wenyin, J. Zhai, D. Dori, Extended summary of the arc segmentation contest, in *Graphics Recognition Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 343–349

74. D. Elliman, Tif2vec, an algorithm for arc segmentation, in *engineering drawings, in Graphics Recognition Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 350–358
75. X. Hilaire, RANVEC and the Arc Segmentation Contest, in *Graphics Recognition - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 359–364
76. A.K. Chhabra, I.T. Phillips, The second international graphics recognition contest—raster to vector conversion: a report, in Tombre, Chhabra [100], pp. 390–410
77. I.T. Phillips, J. Liang, A.K. Chhabra, R.M. Haralick, *A performance evaluation protocol for graphics recognition systems* (Algorithms and Systems, Second International Workshop, Selected Papers, Graphics Recognition, 1997), pp. 372–389
78. R. Kasturi, K. Tombre, Summary and recommendations, *Graphics Recognition, Methods and Applications, First International Workshop, University Park, PA, USA, August 10-11, 1995, Selected Papers* (1995), pp. 301–308
79. D. Dori, L. Wenyin, M. Peleg, How to win a dashed line detection contest, in Kasturi, Tombre [101], pp. 286–300
80. B. Kong, I.T. Phillips, R.M. Haralick, A. Prasad, R. Kasturi. A benchmark: performance evaluation of dashed-line detection algorithms, in Kasturi, Tombre [101], pp. 270–285
81. K. Tombre, Is graphics recognition an unidentified scientific object?, *Graphics Recognition. Recent Advances and New Opportunities, 7th International Workshop, GREC 2007, Selected Papers* (2007), pp. 329–334
82. B. Lamiroy, D.P. Lopresti, H.F. Korth, J. Heflin, How carefully designed open resource sharing can help and expand document analysis research, in *Document Recognition and Retrieval XVIII, part of the IS&T-SPIE Electronic Imaging Symposium* (2011), p. 78740O
83. B. Lamiroy, D.P. Lopresti, An open architecture for end-to-end document analysis benchmarking, in *International Conference on Document Analysis and Recognition* (2011), pp. 42–47
84. M. Tooley, D. Wyatt, *Aircraft engineering principles and practice, Aircraft Electrical and Electronic Systems: Principles, Operation and Maintenance* (Butterworth-Heinemann, Oxford, 2008)
85. K. Tombre, B. Lamiroy, Pattern recognition methods for querying and browsing technical documentation, *Progress in Pattern Recognition, Image Analysis and Applications, 13th Iberoamerican Congress on Pattern Recognition* (2008), pp. 504–518
86. J.-P. Salmon, Reconnaissance de symboles complexes. Ph.D. thesis, Institut National Polytechnique de Lorraine, 2008
87. FRESH. Final report on symbol recognition with evaluation of performances. Deliverable 2.4.2. - FP6-516059, 2007
88. M. Rusiñol, J. Lladós, *Symbol Spotting in Digital Libraries: Focused Retrieval Over Graphic-Rich Document Collections* (Springer, London, 2010)
89. M.M. Luqman, *Fuzzy multilevel graph embedding for recognition, indexing and retrieval of graphic document images*. Ph.D. thesis, Francois Rabelais University of Tours France and Autonoma University of Barcelona Spain, 2012
90. N. Nayef. *Geomatric-based symbol spotting and retrieval in technical line drawings*. Ph.D. thesis, University of Kaiserslautern, Germany, 2012
91. E. Valveny, S. Tabbone, O. Ramos, E. Philippot, *Performance characterization of shape descriptors for symbol representation, Graphics Recognition*, Lecture Notes in Computer Science Series (Springer, Berlin, 2007), pp. 278–287
92. J. Rendek, G. Masini, Ph Dosch, K. Tombre, The search for genericity in graphics recognition applications: design issues of the qgar software system, in *Proceedings of the 6th IAPR International Workshop on Document Analysis Systems, Florence, (Italy).* Lecture Notes in Computer Science, vol. 3163 (2004), pp. 366–377
93. S. Tabbone, L. Wendling, D. Zuwala, A hybrid approach to detect graphical symbols in documents, in *Proceedings of International Workshop on Document Analysis Systems*, ed. S. Marinai, A. Dengel. Lecture Notes in Computer Science (Springer, Berlin, 2004), pp. 342–353

94. R.J. Qureshi, J.-Y. Ramel, D. Barret, H. Cardot, Spotting symbols in line drawing images using graph representations, in *Proceedings of IAPR International Workshop on Graphics Recognition*, ed. by W. Liu, J. Lladós, J.-M. Ogier. Lecture Notes in Computer Science, vol. 5046 (Springer, Berlin, 2008), pp. 91–103
95. H. Bunke, P.S.P. Wang (eds.), *Handbook of Character Recognition and Document Image Analysis* (World Scientific, Singapore, 1997)
96. L.P. Cordella, M. Vento, Symbol recognition in documents: a collection of techniques? Int. J. Doc. Anal. Recognit. **3**(2), 73–88 (2000)
97. M. Robinson, L.S. Baum, J.H. Boose, D.B. Shema, S.C. Chew, Case study: boeing intelligent graphics for airplane operations and maintenance, in *Conference XML'98*, Chicago, USA, 1998
98. S. Adam, J.M. Ogier, C. Cariou, R. Mullot, J. Labiche, J. Gardes, Symbol and character recognition: application to engineering drawings. Int. J. Doc. Anal. Recognit. **3**(2), 89–101 (2000)
99. L. Baum, J. Boose, M. Boose, C. Chaplin, R. Provine, Extracting system-level understanding from wiring diagram manuals, in *Proceedings of 5th IAPR International Workshop on Graphics Recognition, Barcelona (Spain)* (2003), pp. 132–138
100. K. Tombre, A.K. Chhabra (eds.), *Graphics Recognition-Algorithms and Systems*, vol. 1389 (Lecture Notes in Computer Science (Springer, Berlin, 1998)
101. R. Kasturi, K. Tombre (eds.), *Graphics Recognition-Methods and Applications*, vol. 1072 (Lecture Notes in Computer Science (Springer, Berlin, 1996)

Chapter 3
Graphics Recognition and Validation Protocol

3.1 Basic Steps: Symbol Recognition Systems

As mentioned in Sects. 2.1 and 2.2 of Chap. 2, graphics recognition refers to the the recognition of graphical symbols or any meaningful shapes. The research topic has been the subject of numerous reviewed research articles [1–7]. More often, existing systems are composed of two major units:

(a) data acquisition and preprocessing; and
(b) data representation and recognition.

Keeping these items in mind, in the following, a brief review will be made.

3.1.1 Data Acquisition and Preprocessing

Scanned documents at grayscale often include noise, distortion, and deformation at different levels. Broadly speaking, in raster data, acquisition and preprocessing tools/techniques are aimed at retrieving the input data to the most "relevant" information in accordance with the need. This helps locate the regions-of-interest in any studied document image. Out of several different steps, a binarization step ([8, 9], for example) can be used to retrieve important information. It is important to note that the binarization process influences the subsequent processes, which are related to data representation and recognition techniques. As an example, in the framework of binarization, foreground and background separation can be considered as an important tool [10]. Not to be confused, foreground and background separation cannot be taken as a generic tool for all types of applications. This means that, in high-level term, document segmentation is required and it varies from one application to another [11]. For example, inconsistent results happen from issues, such as defects such as creases, stains on paper and heterogeneous wears. Similar issues have to be corrected by taking specific operators (algorithms) into account. This can be considered

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_3

as a precursor for any suitable data acquisition and preprocessing. In some cases, human expertise, i.e., user intervention, is required in addition to the automatically operated tools/techniques.

Other challenging issues are how to define criteria and their evaluation (protocol) for the quality document interpretation, since definition varies from one application to another. In other words, output from this unit may undergo subsequent processes that are consistent with the sentiments of the data and/or the needs. For example, if we consider an application: text/graphics separation [12–18], we will be able to prune the data by removing small connected components. In other words, for such an application, small connected components refer to texts and their removal help focus on graphics. Basically, text/graphics separation means decomposing document image into two layers. For more detailed information about the usefulness of text/graphics separation, we can refer to previous works [19]. Several different techniques were used to separate text parts from background [20]. Among them, Fletcher and Kasturi [21] provided promising results and can be considered for wide ranges of data. The primary issue, however, is that it has to rely on a priori knowledge (the size of connected components, for instance), since it may omit small disconnected graphical symbols/shapes/elements. In such a case, one has to consider local segmentation that helps advance performance. For this problem, recently, sparse representation has been used [22].

On the whole, take home message is that the unit we call acquisition and preprocessing requires a priori knowledge about the document structure the complexity of the graphics in there. Knowing both the layout of the document and the complexity of the graphics help the right use of document segmentation techniques and the subsequent processes.

3.1.2 Data Representation and Recognition

As mentioned earlier, the quality preprocessor yields quality data at the output and data representation and recognition steps are related to what has been processed before.

In Chap. 2, it was mentioned that data, i.e., graphical symbols are represented either in terms of feature vectors that estimates the overall shape (statistically speaking) or in more structured forms (graphs, for instance: structurally speaking) by taking meaningful visual cues, element or primitives. In structural approach, the primary part is how can we select tools that can extract visual cues, visual elements and/or primitives that can convey overall shape/structure of the graphical symbol. Note that, visual primitive selection tools vary in accordance with the requirement, i.e., they are application dependent.

It is important to observe that data quality and their complexity help determine how data can be represented. In a similar fashion, data representation is followed by matching techniques that are used in the decision process. This means that an appropriate data representation is assumed to be compact and discriminant. In addition, the data representation (feature vector, for instance) is expected to minimize the intra-class distance and maximizes the inter-class distance [23]. Intra-class distance is measured within a specific class and similarly, the distance between two samples belonging to two different classes refers to inter-class distance. Further, feature computation must be practically feasible in terms of time complexity as well as the data portability, i.e., storage format. Considering both issues, classification task can be easier.

Recognition/classification is considered as the process of identifying to which class a test sample/data belongs. The performance is based on how big is the training data. Not just in graphics recognition, in general, automatic recognition, description, classification and grouping of patterns (clustering) are important issues in many disciplines, such as biology, chemistry, physics and remote sensing, where data, i.e., pattern/signal representation (shape analysis, for example) plays a crucial role [6, 24]. Patterns can be either graphical symbols [25] or other visual structures like cursive characters [26]. For all kinds of pattern, broadly, following the state-of-the-art literature, for pattern representation, any one or the combination of the following three different approaches can be used:

(a) statistical [27–30],
(b) structural [31–33] and
(c) syntactic [34–36].

The selection of the approach relies on the application, i.e., problem (data) complexity [37, 38]. In Chaps. 4–6, a detailed explanation will be provided. In the literature, several methods are particularly worked on isolated symbols. This means that complex and composite symbols connected to a complex environment (texts, for instance) were been sidelined [5, 23, 25, 39]. Let us summarize reasons with examples.

(a) The techniques/algorithms that come under statistical approaches fairly check the similarity based on the computed distances between two feature vectors [25, 40]. For graphical symbol recognition problem, let us refer the the previous work, which is based on Dynamic Programming (DP) technique to match the Radon signatures [41–43].
(b) To handle vector-based representation of primitives (attributed relational graphs (ARG), for example) [5, 44–46], graph matching techniques/algorithms that come under structural approaches are effective. In case nodes are labeled, graph matching can be straightforward, i.e., relational signature alignment [47–53]. This can definitely avoid NP-hard problem that can always happen in general graph matching techniques.
(c) The techniques/algorithms based on graph grammar that come under syntactic approaches will be appropriate to search graphical elements/symbols or meaning parts/regions in technical documents [34–36, 54–57].

Note that graph grammar may be close to a feature vector description that follows composition rules of visual primitives. Besides, user-intervention (or expert) can help advance any of the approaches mentioned above.

In a few words, we have neither observed any absolute standard on the choice of the best approaches, nor found straight forward combinations of techniques/algorithms.

More often, structural and statistical pattern recognition approaches have been integrated [58]. In a similar fashion, unification of syntactic and statistical pattern recognition approaches could be an interesting concept [59]. It holds true for syntactic and structural pattern recognition approaches [38, 60–63].

3.2 Validation

To make a fair comparison (often called, an apple-to-apple comparison), one should note the following two majors points:

(a) Datasets; and
(b) Evaluation metrics.

This means that, in order to see, how far we have been advanced, one needs to follow the exact similar evaluation protocol. More often, the characteristics of the datasets, their availability for further researches and the applications (or intentions) may change one's evaluation metric. Not just datasets and evaluation metrics, we may be biased in implementing previously reported algorithms. As a consequence, we are unable to track researches done over several years, since results cannot be consistent as algorithms may not be tuned (i.e., parameters) as in the original references [64].

In the year 2010, Lehigh University targeted at the Document Image Analysis (DIA) research community for distributing document images and associated document analysis algorithms. In addition, it provides an unlimited range of annotations and "ground truth" for benchmarking and evaluation of new contributions to the state-of-the-art [65]. For more detailed information of the DAE project, one can follow the link: (URL: http://dae.cse.lehigh.edu/DAE/), where the goal is written as follows:

> *Building and maintaining a national resource to support critical research and development in translation, document analysis, preservation, and exploitation.*

Besides, it has been clearly mentioned that the DAE project aims to provide an environment for the hosting and distribution of reproducible data, annotations and software [66–68]. As said in [68], DAE was conceived and built around a core data model that establishes an exhaustive range of relations between document images, annotation areas, interpretations or ground truth, but also links the data to user interactions, experimental protocols or program executions. It also provides several different services, such as querying, up- and download and remote execution.

Of course, having new datasets will help us move forward. This will not certainly tell us that we have improved with respect to the literature or previous works.

3.2.1 Datasets, Evaluation Protocol, and Their Relation

In [69], Haralick mentioned that performance evaluation for the DIA has a different emphasis than performance evaluation in other research areas. Following Sect. 2.3 of Chap. 2, we have observed that different datasets and evaluation protocols have been provided since 1995. Besides, evaluation metrics are varied not just because of the datasets but also because of the intended applications.

A quick overview of the performance evaluation protocol for graphics recognition systems can be found in the work reported by Phillips et al. [70]. In their work, authors define a computational protocol for evaluating the performance of raster to vector conversion systems, since their graphical entities handled are continuous and dashed lines, arcs, and circles, and text regions. In addition, their protocol allows matches of the type one-to-one, one-to-many, and many-to-one between the ground truth and the test results. As mentioned earlier, it is important to note that datasets are important, and in some cases, authors generated synthetic data and ground truths [71, 72], and at the same time we are aware of applications: symbol recognition and spotting systems. Authors applied their approach to generate large data (relatively) of architectural drawings and electrical circuit diagrams so that it provides flexibility to realize well-trained system. For more information, we can refer to other works [73–76]. In a similar fashion, for arc segmentation [77, 78], authors proposed different idea on their evaluation protocol.

In short, evaluation metrics can be changed and it is mainly based on the complexity of associated datasets, ground truths and applications. In what follows, few evaluation metrics are provided with appropriate arguments.

3.2.2 Evaluation Metric

Let us assume feature-based matching between two images: query (q) and database (d) image.

Matching score between two patterns tells us how similar or dissimilar they are. For any query pattern $\{\mathscr{P}^q\}_{q=1,\ldots,Q}$ over all database (or template) patterns $\{\mathscr{D}^d\}_{d=1,\ldots,D}$, distance vector can be computed as,

$$\text{Distance}(\mathscr{P}^q, \mathscr{D}^d) = \begin{bmatrix} S^{q,1} \\ S^{q,1} \\ \vdots \\ S^{q,D} \end{bmatrix},$$

where matching score S is based on the distance between two patterns: $S^{q,d} = \delta(\mathscr{P}^q, \mathscr{P}^d) = \sum ||\mathscr{I}_i^q - \mathscr{I}_i^d||$, for instance. We must be aware of computing distance between two candidates is depend on what metric we use, and in our case, we used l_2-norm. In general, matching techniques are influenced by how we represent patterns. One has to understand how to identify the appropriate use of the evaluation metric, since, more often, it follows nature of the datasets/application.

To generalize similarity between the two given test candidates, similarity score can be normalized into [0, 1] by,

$$\overline{S} = \frac{S - S_{\min}}{S_{\max} - S_{\min}}.$$

As a result, for a better understanding, similarity can formally be expressed as,

$$\begin{aligned} \text{Similarity}(\mathscr{P}^q, \mathscr{P}^d) &= 1 - \overline{Distance}(\mathscr{P}^q, \mathscr{P}^d) \\ &\equiv \begin{cases} 1 & \text{the closest pattern/candidate} \\ 0 & \text{the farthest pattern/candidate.} \end{cases} \end{aligned}$$

This means that database patterns are ranked in accordance with the decreasing order of similarity in [1, 0].

Example 1.

Consider a query pattern $\mathscr{P}^q$ that is matched with database pattern: $\{\mathscr{P}^d\}_{d=1,\ldots,10}$, the corresponding distance vector can be expressed as,

$$\text{Distance}(\textbf{.}) = \begin{bmatrix} 10 \\ 8 \\ 2 \\ 1 \\ 3 \\ 4 \\ 7 \\ 8 \\ 9 \\ 5 \end{bmatrix},$$

then $S_{\min} = 1$ and $S_{\max} = 10$, and

$$\overline{\text{Distance}}(\textbf{.}) = \begin{bmatrix} 1 \\ 7/9 \\ 1/9 \\ 0 \\ 2/9 \\ 3/9 \\ 6/9 \\ 7/9 \\ 8/9 \\ 4/9 \end{bmatrix}.$$

Now,

$$\text{Similarity}(\textbf{.}) = \begin{bmatrix} 0 \\ 2/9 \\ 8/9 \\ 1 \\ 7/9 \\ 6/9 \\ 3/9 \\ 2/9 \\ 1/9 \\ 5/9 \end{bmatrix}.$$

Based on Similarity(.) score, one can consider the following measures: recognition and retrieval. In other words, these metrics can be used according to what we are looking at; it can be either recognition, spotting or retrieval.

3.2.3 Recognition

Recognition can be considered as the straightforward nearest neighbor algorithm that can classify/detect the closest candidate [79]. In other words, the query/test pattern is said to be recognized with the database pattern from which it produces the smallest possible score. It can be expressed as,

$$\text{Recognition}^d = \underset{d \in D}{\text{argmin}} \begin{bmatrix} S^{q,1} \\ S^{q,1} \\ \vdots \\ S^{q,D} \end{bmatrix}.$$

In case of normalized distance vector, Similarity$(\textbf{.}) = 1$ for the matched/recognized pattern. Therefore, following the previous example, pattern in *index 4* is the matched pattern for that specific query. For a quick understanding, one can considered it as the k Nearest Neighbor (NN) classification problem, where $k = 1$.

3.2.4 Retrieval

Not just relying on the best candidate/pattern, search space can be increased to select other potential similar candidates for the query. This allows to produce a short-list from the database based on the similarity scores. There are several different ways to evaluate retrieval performance, and they are dataset dependent and ground truths. They are in general,

(a) balanced dataset and
(b) imbalanced dataset.

Considering the ground truth information, real-world datasets are not labeled nor do they are balanced. This makes evaluation difficult in terms of a fair comparison with the state-of-the-art approaches, since the evaluation metrics can be different from now and then. Researchers often selected (manually) data to make the right fit (ground truths) so that comparison is possible. To avoid possible biasing for a comparison, much efforts were been added [80].

Based on the nature of the datasets, in the following, two different types of retrieval metrics can be summarized.

3.2.4.1 Fully Labeled and Balanced Dataset

Balanced dataset is expected to have identical numbers of ground truth for all known classes. For example, shape datasets *kimia shape99* [81] consists of 9 classes, each one is having 11 samples. In such a case, we use mainly two different retrieval measures, and can be made fair comparison with all previously reported works.

1. *Precision* and *recall*

   For any chosen query, conventional *precision* and *recall* measures can be explained as follows. Consider a query, precision can be computed as,

   $$\text{Precision} = \frac{n}{k},$$

   where n is the number of retrieved relevant candidates and k is the number of retrieval candidates. In a similar fashion, *recall* can be computed as,

   $$\text{Recall} = \frac{n}{N},$$

   where N is the number of relevant candidates, which we call ground truths.

   Considering performance evaluation, *recall* alone is not enough since it is trivial to produce *recall* of 100% by returning all candidates in response to any query. Therefore, computing the *precision* is more important to see how many nonrelevant candidates are retrieved.

   Fscore is another measure, which combines *precision* and *recall*. It can be expressed as,

   $$\text{Fscore} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.$$

Since *recall* and *precision* are evenly/equally weighted, we can call it as Fscore_1 measure.

In general, Fscore_β measure (for nonnegative real values of β) can be expressed as,

$$\text{Fscore}_\beta = (1+\beta^2) \times \frac{\text{precision} \times \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}.$$

Note that two commonly used F measures are the F_2 measure and $F_{0.5}$ measure. The former one weights *recall* twice as much as *precision* and the latter one weights *precision* twice as much as *recall*.

As a reminder, *precision* takes all retrieved images into account. What if we can retrieve and evaluate an algorithm at a given cut-off rank? In other words, can we consider only the top-most results returned by the algorithm? Such a measure is called by *precision* at K or $precision@k$ since $k = 1, \ldots, K$, where K is the requested list or cut-off. Having such a framework, *retrieval rate* can be known by another measure called *retrieval accuracy* [79, 81, 82]. It is defined as the ratio of the number of correctly classified candidates to the requested lists.

2. *Bull's eye score*

In this metric/measure, the idea is to increase the search space and it can go up to 2 times the number of relevant patterns/candidates in the dataset for each studied class.For such a search space, test sample is compared with all candidates (patterns in the database) and the numbers of correctly classified pattern/candidate are reported. *Bull's eye score* [83–86] can be expressed as,

$$\text{Bull's eye score} = \frac{\sum n}{N}.$$

Like before, n is the number relevant patterns/candidates and N, ground truth for the specific query.

Example

Let us have a query $Q1$ with 10 ground truths, i.e., $N = 10$ and therefore, the shortlist we request is $K = 10$. For a test, the boolean result is as follow:

$$\text{Output}_{Q1} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

where 1 refers to correctly classified pattern/candidate and 0, otherwise. Then, the *precision* and *recall* are

$$\text{Precision}_{Q1} = \begin{bmatrix} 1/1 \\ 2/2 \\ 2/3 \\ 3/4 \\ 3/5 \\ 4/6 \\ 5/7 \\ 6/8 \\ 6/9 \\ 6/10 \end{bmatrix} \text{ and Recall}_{Q1} = \begin{bmatrix} 1/10 \\ 2/10 \\ 2/10 \\ 3/10 \\ 3/10 \\ 4/10 \\ 5/10 \\ 6/10 \\ 6/10 \\ 6/10 \end{bmatrix}.$$

Fscore is fairly straightforward as soon as we have *precision* and *recall* values. At this point, global/average *retrieval accuracy* or *retrieval rate* is the last element of the *precision* vector as mentioned above. Since the shortlist is taken as the value of ground truths, i.e., $K = N$, and in this example, one can see that *recall* is limited to 60%. Increasing the shortlist can increase the *recall* but, *precision* can be decreased.

In *bull's eye test*, as said earlier, $K = 2 \times N$. In this case, retrieval result can be expressed as,

$$\text{Output}_{Q1} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

where, *bull's eye score* $= 100\%$.

3.2.4.2 Imbalanced Dataset

Real-world data are often imbalanced. Imbalanced data may not have identical numbers of ground truths for all classes. In this context, *retrieval efficiency* [49, 87] may be an appropriate metric, since *precision* and *recall* curve could potentially be biased. *Retrieval efficiency* for a shortlist of size K can be computed as,

$$\eta_K = \begin{cases} n/N & \text{if } N \leq K \\ n/K & \text{otherwise,} \end{cases}$$

where, like before, n is the number of returned relevant candidates and N the total number of relevant ones in the dataset. Note that, if $N \leq K$, η_K computes conventional *recall*. In an opposite condition, it computes *precision*. In this way, the output contains similar patterns/candidates in terms of both precision or recall, for any provided shortlist.

As mentioned above, one of the key merits of the metric is that average retrieval efficiency curve cannot biased even if we have different ground truths for different test samples/queries. For more understanding, an example (see below) will help understand the process.

Example

Consider a query $Q1$ has 6 ground truths, i.e., $N = 6$ and we request for $K = 10$. The boolean result is as follow:

$$\text{Output}_{Q1} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

where 1 refers to correctly classified pattern/candidate and 0, otherwise. Then, the *precision* and *recall* are

$$\text{Precision}_{Q1} = \begin{bmatrix} 1/1 \\ 2/2 \\ 2/3 \\ 3/4 \\ 3/5 \\ 4/6 \\ 5/7 \\ 6/8 \\ 6/9 \\ 6/10 \end{bmatrix}, \text{ and } \text{Recall}_{Q1} = \begin{bmatrix} 1/6 \\ 2/6 \\ 2/6 \\ 3/6 \\ 3/6 \\ 4/6 \leftarrow \\ 5/6 \\ 6/6 \\ 6/6 \\ 6/6 \end{bmatrix}.$$

Retrieval efficiency can now be computed as,

$$\text{Retrieval efficiency}_{Q1} = \begin{bmatrix} 1/1 \\ 2/2 \\ 2/3 \\ 3/4 \\ 3/5 \\ 4/6 \\ 5/6 \\ 6/6 \\ 6/6 \\ 6/6 \end{bmatrix}.$$

Note that *precision* has been biased from the index 6 (see arrow). This means that, in *retrieval efficiency*, *recall* is combined with *precision* from where it starts getting biased. Therefore, retrieval efficiency can be considered as the best fit metric in case of imbalanced dataset(s) [49, 50, 52, 53]. Similar works can be found in [88, 89]

In short, two different examples address the appropriate selection of the evaluation metric in accordance with nature of the dataset. In cases of labeled and balanced dataset, *precision* and *recall* measures are trivial. It can, however, be also used in imbalanced dataset if and only if the shortlist is less than or equal to the minimum ground truths, i.e., $K \leq N$ from the set of queries. For instance, if any query Q^q has the smallest number of relevant patterns/candidates, then that value is taken as the maximum short list to compute *precision* and *recall*. Alternatively, *retrieval efficiency* could potentially be the best selection as it combines both of them without biasing.

3.3 Summary

In this chapter, we have quickly explained basic idea symbol recognition systems, where it includes how data can be taken and preprocessed before representing them in a way they can be classified as expected. All graphics recognition systems or algorithms require validation protocol and we have observed that their protocol varied from one application to another. More often, changes happened because of the nature of the datasets. However, reported results were found to be in the forms of recognition, precision, and retrieval. While explaining this, we have also discussed about Document Analysis and Exploitation (DAE), which is hosted at the Lehigh University and the goals.

Computing recognition is straightforward as it happens in all dataset types. However, retrieval measure is not trivial. This means that it needs an attention for an appropriate selection so that a fair comparison can be made with the reported state-of-the-art methods.

As mentioned in Sect. 3.1, to represent a pattern, any one or the combination of the following three different approaches can be used: (i) statistical, (ii) structural and (iii) syntactic. In the following chapters, we will discuss them in detail.

References

1. A.K. Chhabra, Graphic symbol recognition: an overview, in *Proceedings of 2nd International Workshop on Graphics Recognition, Nancy (France)* (1997), pp. 244–252
2. D.S. Doermann, An introduction to vectorization and segmentation, in Tombre, Chhabra [90], pp. 1–8
3. R. Kasturi, R. Raman, C. Chennubhotla, L. O'Gorman, Document image analysis: an overview of techniques for graphics recognition, in *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)* (1990), pp. 192–230
4. A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000). January
5. J. Lladós, E. Martí, J.J. Villanueva, Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. IEEE Trans. Pattern Anal. Mach. Intell. **23**(10), 1137–1143 (2001)
6. S. Loncaric, A survey of shape analysis techniques. Pattern Recognit. **31**(8), 983–1001 (1998)
7. S. Marshall, Review of shape coding techniques. Image Vis. Comput. **7**(4), 281–294 (1989)
8. Ø.D. Trier, T. Taxt, Evaluation of binarization methods for document images. IEEE Trans. Pattern Anal. Mach. Intell. **17**(3), 312–315 (1995)
9. J. Sauvola, M. Pietikäinen, Adaptive document image binarization. Pattern Recognit. **33**(2), 225–236 (2000)
10. U. Garain, T. Paquet, L. Heutte, On foreground-background separation in low quality document images. Int. J. Doc. Anal. Recognit. **8**(1), 47–63 (2006)
11. T. Taxt, P.J. Flynn, A.K. Jain, Segmentation of Document Images. IEEE Trans. Pattern Anal. Mach. Intell. **11**(12), 1322–1329 (1989)
12. S. Ablameyko, O. Okun, Text separation from graphics based on compactness and area properties. Mach. Graph. Vis. **3**(3), 531–541 (1994)
13. H. Luo, R. Kasturi, Improved directional morphological operations for separation of characters from maps/graphics, in Tombre, Chhabra [90], pp. 35–47
14. L. Wenyin, D. Dori, A proposed scheme for performance evaluation of graphics/text separation algorithms, in Tombre, Chhabra [90], pp. 359–371
15. R. Cao, C.L. Tan, Text/graphics separation, in *maps, in Graphics Recognition - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 167–177
16. R. Cao, C.L. Tan, Separation of overlapping text from graphics, in *Proceedings of the 6th International Conference on Document Analysis and Recognition, Seattle, WA (USA)* (2001), pp. 44–48

17. K. Tombre, S. Tabbone, L. Pélissier, B. Lamiroy, Ph. Dosch. Text/graphics separation revisited, in *Proceedings of the 5th IAPR International Workshop on Document Analysis Systems, Princeton, NJ (USA)*, ed. by D. Lopresti, J. Hu, R. Kashi. Lecture Notes in Computer Science, vol. 2423 (Springer, Berlin, 2002), pp. 200–211
18. A. Velázquez, S. Levachkine, Text/graphics separation and recognition in raster-scanned color cartographic maps, in *Proceedings of 5th IAPR International Workshop on Graphics Recognition, Barcelona (Spain)* (2003), pp. 92–103
19. S. Tabbone, L. Wendling, K. Tombre, Matching of graphical symbols in line-drawing images using angular signature information. Int. J. Doc. Anal. Recognit. **6**(2), 115–125 (2003)
20. D. Doermann, K. Tombre, *Handbook of Document Image Processing and Recognition* (Springer, New York Incorporated, 2014)
21. L.A. Fletcher, R. Kasturi, A robust algorithm for text string separation from mixed text/graphics images. IEEE Trans. Pattern Anal. Mach. Intell. **10**(6), 910–918 (1988)
22. T.H. DO, Sparse representation over learned dictionary for document analysis. Ph.D. thesis, LORIA, Université de Lorraine, France (2014)
23. J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *Graphics Recognition - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 104–127
24. D. Zhang, G. Lu, Review of shape representation and description techniques. Pattern Recognit. **37**(1), 1–19 (2004)
25. L.P. Cordella, M. Vento. Symbol and shape recognition, in *Proceedings of 3rd International Workshop on Graphics Recognition, Jaipur (India)* (1999), pp. 179–186
26. S. Watanabe, *Pattern Recognition: Human and Mechanical* (Wiley, New York, 1985). ISBN 0471808156
27. J. Kittler, Statistical pattern recognition: the state of the art, in *Image Analysis and Processing*, ed. by V. Cantoni, V. Di Gesù, S. Levialdi, vol. 2, (Plenum Press, New York, 1987), pp. 57–66
28. S. Raudys, A.K. Jain, Small sample size effects in statistical pattern recognition: recommandations for practitioners. IEEE Trans. Pattern Anal. Mach. Intell. **13**(1), 252–264 (1991)
29. K. Fukunaga, Statistical pattern recognition, in Chen et al. [91], Chap. 1.2, pp. 33–60
30. A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
31. T. Pavlidis, *Struct. Pattern Recognit.* (Springer, New York, 1980)
32. W.J. Christmas, J. Kittler, M. Petrou, Structural matching in computer vision using probabilistic relaxation. IEEE Trans. Pattern Anal. Mach. Intell. **17**(8), 749–764 (1995)
33. J. Lladós, E. Martí, Structural recognition of hand drawn floor plans, in *6th Spanish Symposium on Pattern Recognition and Image Analysis, Cordoba* (1995), pp. 27–34
34. V. Claus, H. Ehrig, G. Rozenberg (eds.), *Graph-Grammars and Their Applications to Computer Science and Biology (Lecture Notes in Computer Science* (Springer, Berlin, 1979)
35. D. Dori, A. Pnueli, The grammar of dimensions in machine drawings. Comput. Vis. Graph. Image Process. **42**, 1–18 (1988)
36. M. Flasiński, Characteristics of edNLC-Graph Grammar for Syntactic Pattern Recognition. Comput. Vis. Graph. Image Process. **47**, 1–21 (1989)
37. H. Bunke, A. Sanfeliu (eds.), *Syntactic and Structural Pattern Recognition* (World Scientific, Singapore, 1990)
38. H. Bunke, Structural and syntactic pattern recognition, in Chen et al. [91], Chap. 1.5, pp. 163–209
39. L.P. Cordella, M. Vento, Symbol recognition in documents: a collection of techniques? Int. J. Doc. Anal. Recognit. **3**(2), 73–88 (2000)
40. S. Müller, G. Rigoll, Engineering drawing database retrieval using statistical pattern spotting techniques, in *Proceedings of 3rd International Workshop on Graphics Recognition, Jaipur (India)* (1999), pp. 219–226
41. K.C. Santosh, B. Lamiroy, L. Wendling, Dtw-radon-based shape descriptor for pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **27**(3), 1350008 (2013)

42. K.C. Santosh, Character recognition based on dtw-radon, in *Proceedings of International Conference on Document Analysis and Recognition* (2011), pp. 264–268
43. K.C. Santosh, B. Lamiroy, L. Wendling, DTW for matching radon features: a pattern recognition and retrieval method, in *Advances Concepts for Intelligent Vision Systems (ACIVS)* (2011), pp. 249–260
44. B.T. Messmer, H. Bunke, Efficient error-tolerant subgraph isomorphism detection, in *Shape, Structure and Pattern Recognition (Post-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition Nahariya, Israel)*, ed. by D. Dori, A. Bruckstein (World Scientific, Singapore, 1995), pp. 231–240
45. J.-Y. Ramel, G. Boissier, H. Emptoz. A structural representation adapted to handwritten symbol recognition, in *Proceedings of 3rd International Workshop on Graphics Recognition, Jaipur (India)* (1999), pp. 259–266
46. J.Y. Ramel, N. Vincent, H. Emptoz, A structural representation for understanding line-drawing images. Int. J. Doc. Anal. Recognit. **3**(2), 58–66 (2000)
47. K.C. Santosh, L. Wendling, B. Lamiroy, Using spatial relations for graphical symbol description, in *Proceedings of the IAPR International Conference on Pattern Recognition* (IEEE Computer Society, 2010), pp. 2041–2044
48. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2011), pp. 228–237
49. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis). Ph.D. thesis, University of Lorraine, France (2011)
50. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
51. K.C. Santosh, L. Wendling, B. Lamiroy, Relation bag-of-features for symbol retrieval, in *12th International Conference on Document Analysis and Recognition* (2013), pp. 768–772
52. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
53. K.C. Santosh, L. Wendling, Bor: bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. Artif. Intell. **28**(06), 1450017 (2014)
54. W.H. Tsai, K.S. Fu, Attributed grammar: a tool for combining syntactic and statistical approaches to pattern recognition. IEEE Trans. Syst. Man Cybern. **10**(12), 873–885 (1980)
55. K.C. You, K.S. Fu, Distorted shape recognition using attributed grammars and error-correcting techniques. Comput. Vis. Graph. Image Process. **13**, 1–16 (1980)
56. L.P. Cordella, P. Foggia, R. Genna, M. Vento, Prototyping structural descriptions: an inductive learning approach, in *Advances in Pattern Recognition (Proceedings of Joint IAPR Workshops SSPR'98 and SPR'98, Sydney, Australia)*, ed. by A. Amin, D. Dori, P. Pudil, H. Freeman. *Lecture Notes in Computer Science*, vol. 1451 (1998), pp. 339–348
57. K.C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2009), pp. 1330–1334
58. H.S. Baird, Feature identification for hybrid structural/statistical pattern classification. Comput. Vis. Graph. Image Process. **42**, 318–333 (1988)
59. K.S. Fu, A step towards unification of syntactic and statistical pattern recognition. IEEE Trans. Pattern Anal. Mach. Intell. **5**(2), 200–205 (1983)
60. L. Miclet, Grammatical inference, in *Syntactic and Structural Pattern Recognition: Theory and Applications (Chap. 9)*, ed. by H. Bunke, A. Sanfeliu (World Scientific, Singapore, 1990), pp. 237–290
61. S. Satoh, T. Satou, M. Sakauchi, One Method of Structural Description Rule Extraction based on Graphical and Spatial Relations. **2**, 281–284 (1992)
62. K. Tombre, Structural and syntactic methods in line drawing analysis: to which extent do they work? in *Advances in Structural and Syntactial Pattern Recognition (Proceedings of 6th International SSPR Workshop, Leipzig, Germany)*, ed. by P. Perner, P. Wang, A. Rosenfeld. *Lecture Notes in Computer Science*, vol. 1121 (Springer, Berlin, 1996), pp. 310–321

63. J. Lladós, G. Sánchez, E. Martí, A string based method to recognize symbols and structural textures in architectural plans, in *Proceedings of 2nd International Workshop on Graphics Recognition, Nancy (France)* (1997), pp. 287–294
64. B. Lamiroy, D.P. Lopresti, H.F. Korth, J. Heflin, How carefully designed open resource sharing can help and expand document analysis research, in *Document Recognition and Retrieval XVIII, Part of the IS&T-SPIE Electronic Imaging Symposium* (2011), p. 78740O
65. B. Lamiroy, D.P. Lopresti, An open architecture for end-to-end document analysis benchmarking, in *2011 International Conference on Document Analysis and Recognition* (2011), pp. 42–47
66. B. Lamiroy, D.P. Lopresti, The non-geek's guide to the DAE platform, in *10th IAPR International Workshop on Document Analysis Systems* (2012), pp. 27–32
67. B. Lamiroy, D.P. Lopresti, The DAE platform: a framework for reproducible research in document image analysis, in *Reproducible Research in Pattern Recognition - First International Workshop, RRPR@ICPR 2016, Revised Selected Papers* (2016), pp. 17–29
68. B. Lamiroy, DAE-NG: a shareable and open document image annotation data framework, in *1st International Workshop on Open Services and Tools for Document Analysis, 14th IAPR International Conference on Document Analysis and Recognition* (2017), pp. 31–34
69. R.M. Haralick, Performance evaluation of document image algorithms, in *Graphics Recognition-Recent Advances*, ed. by A.K. Chhabra, D. Dori, Lecture Notes, in Computer Science, vol. 1941, (Springer, Berlin, 2000), pp. 315–323
70. Ihsin T. Phillips, Jisheng Liang, Atul K. Chhabra, Robert M. Haralick, A performance evaluation protocol for graphics recognition systems, in *Algorithms and Systems*, ed. by Graphics Recognition (Second International Workshop, Selected Papers, 1997), pp. 372–389
71. M. Delalandre, J.-Y. Ramel, N. Sidere, A semi-automatic groundtruthing framework for performance evaluation of symbol recognition and spotting systems. Lecture Notes in Computer Science, vol. 7423 (Springer, Berlin, 2013), pp. 163–172
72. M. Delalandre, E. Valveny, T. Pridmore, D. Karatzas, Generation of synthetic documents for performance evaluation of symbol recognition & spotting systems. Int. J. Do. Anal. Recognit. **13**(3), 187–207 (2010)
73. M. Delalandre, E. Valveny, J. Lladós, Performance evaluation of symbol recognition and spotting systems: an overview, in *Proceedings of International Workshop on Document Analysis Systems* ed. by K. Kise, H. Sako (IEEE Computer Society, 2008), pp. 497–505
74. M. Delalandre, J.-Y. Ramel, E. Valveny, M.M. Luqman, A performance characterization algorithm for symbol localization, in *Graphics Recognition. Achievements, Challenges, and Evolution, 8th International Workshop, Selected Papers* (2009), pp. 260–271
75. E. Valveny, M. Delalandre, R. Raveaux, B. Lamiroy, Report on the symbol recognition and spotting contest, in *Graphics Recognition. New Trends and Challenges - 9th International Workshop, Revised Selected Papers* (2011), pp. 198–207
76. FRESH. Final report on symbol recognition with evaluation of performances. Deliverable 2.4.2. - FP6-516059 (2007)
77. W. Liu, D. Dori, Incremental arc segmentation algorithm and its evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **20**(4), 424–431 (1998)
78. S. Song, M.-R. Lyu, S. Cai, Effective multiresolution arc segmentation: algorithms and performance evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **16**(11), 1491–1506 (2004)
79. K.S. Beyer, J. Goldstein, R. Ramakrishnan, U. Shaft, When is "nearest neighbor" meaningful? in *Proceedings of International Conference on Database Theory* (Springer, London, 1999), pp. 217–235
80. E.H. Barney Smith, An analysis of binarization ground truthing, in *Proceedings of International Workshop on Document Analysis Systems* (ACM, New York, 2010), pp. 27–34
81. T.B. Sebastian, P.N. Klein, B.B. Kimia, Recognition of shapes by editing shock graphs, in *Proceedings of International Conference on Computer Vision* (2001), pp. 755–762
82. J. Rendek, L. Wendling, On determining suitable subsets of decision rules using choquet integral. Int. J. Pattern Recognit. Artif. Intell. **22**(2), 207–232 (2008)

83. L.J. Latecki, R. Lakmper, U. Eckhardt, Shape descriptors for non-rigid shapes with a single closed contour, in *Computer Vision and Pattern Recognition* (2000), pp. 1424–1429
84. S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 509–522 (2002)
85. C. Grigorescu, N. Petkov, Distance sets for shape filters and shape recognition. IEEE Trans. Image Process. **12**(10), 1274–1286 (2003)
86. T.B. Sebastian, P.N. Klein, B.B. Kimia, On aligning curves. IEEE Trans. Pattern Anal. Mach. Intell. **25**(1), 116–125 (2003)
87. B.M. Mehtre, M.S. Kankanhalli, A.D. Narasimhalu, G.C. Man, Color matching for image retrieval. Pattern Recognit. Lett. **16**(3), 325–331 (1995)
88. B. Lamiroy, T. Sun, Computing precision and recall with missing or uncertain ground truth, in *Graphics Recognition. New Trends and Challenge*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2013), pp. 149–162
89. B. Lamiroy, On the limits of machine perception and interpretation. (Sur les limites de la perception artificielle et de l'interprétation) (2013)
90. K. Tombre, A.K. Chhabra (eds.), *Graphics Recognition—Algorithms and Systems*. Lecture Notes in Computer Science, vol. 1389 (Springer, Berlin, 1998)
91. C.H. Chen, L.F. Pau, P.S.P. Wang (eds.), *Handbook of Pattern Recognition and Computer Vision* (World Scientific, Singapore, 1993)

Chapter 4
Statistical Approaches

4.1 Statistical Pattern Recognition: A Quick Review

In statistical pattern recognition, appearance-based processing/analysis (shape analysis [1, 2]) plays an important role. This holds true for all pattern recognition problems, and, of course, graphical symbol recognition is a part of it [3, 4]. A quick overview can be found in [5–7], and more importantly, this chapter is motivated by them. Not surprisingly, primary concepts will be taken in such a way that the chapter will be complete by itself.

Shape features are usually categorized into two families [8]:

(a) contour-based descriptors and
(b) region-based descriptors.

Based on the nature or complexity of the studied sample, i.e., pattern one can select one of them or both. However, we must be aware of the following issues:

(a) pattern representation (can be isolated or complex/composite pattern);
(b) matching/comparison between two representations (for recognition purpose); and
(c) extensibility.

The first two items are related to the recognition performance of the algorithm(s). Recognition performance does not just include recognition rate or efficiency but also computational complexity (or processing time). In other words, as mentioned in the earlier chapter, the quality of the representation determines the former factor while matching the latter one.

Briefly, in statistical pattern recognition approaches and considering feature-based (vector, for instance) algorithm, two issues are primarily involved [5, 9]:

(a) feature selection; and
(b) model selection

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_4

for recognition. Note that a pattern is represented as an n-dimensional feature vector, which can mathematically be expressed as $X = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$. The recognition is then made by separating the feature space into known classes. Not to be confused, the paragraph does not convey that feature vectors can only represent patterns; they can be matrices.

In what follows, our discussion will be targeting on how patterns are represented and how can they be benefitting in terms of computational complexity from the way they represent [5].

4.1.1 Contour-Based Shape Analysis

As mentioned before, several approaches are used to describe contours through the use of a small set of features. The feature selection is driven by the needs and/or applications, where issues, such as robustness to noise and tolerance to small distortions can be considered.

Often, to name a few, we have observed frequent use of Fourier descriptor and its variants, polygonal primitives (approximation), curvature-based descriptors, Hough transform, and distance- and angle-based shape descriptor (i.e., shape context).

- Fourier descriptor (FD): It is considered as one of the commonly used techniques to describe shape [10–15]. Because of their simplicity and robustness, such descriptors have been widely used in several different applications, such as character recognition [16].
- Polygonal primitives via contour: It requires dynamic programming for matching, since feature vectors are varied as pattern sizes vary [17, 18]. The approximated/estimated polygonal shape may result in data/information loss. As a consequence, as an example, the degree of ellipticity may not be suitable for polygon recognition [17]). In [19], the authors described a contour-oriented $2D$ pattern, where their recognition is robust to estimate polygonal inconsistency. However, high time complexity can be considered as its major drawback.
- Hough transform (HT) [20–22]: The generalized HT can be considered as another widely used technique. As before, it suffers from high computational time and storage requirement. This means that the work presented in [22] does not process faster and it is practically appropriate. Later, both time and space complexity were reduced with the use of regions-of-interest (ROIs) [23].
- Curvature approaches [24–27]: They describe shapes in scale space by taking contours (boundaries) [24, 25]. Shape similarity can be estimated by computing distance between their corresponding scale-space representations.
  Within this framework, skeleton can be another way to perform pattern matching [28, 29]. Very specifically, for each instance of the pattern, we build a graph by taking the medial axis of the shape into account [29]. Not to be confused, since it aims to extract the skeleton from the shape of the pattern, it is preferred to have silhouette pattern. Extended graphs have been proposed [30–32] with the use of

shock graphs. Other previous works have focused on methods to see how efficient two graphs can be matched [24, 29]. As these methods follow global optimizations, they are found to be effective. On the other hand, they suffer from high complexity issue. In their comparison study, we observed that they may not work on scaling as compared to polar curvature methods [27]. More information about graph matching can be obtained from this chapter.
In just two points, in general,

(a) contour-based descriptors are suitable for silhouette shapes (solid); and
(b) they cannot capture internal contents and may not describe patterns or shapes with hole(s), where boundary is not obvious/clear.

- Shape context (SC): SC is another important descriptor in the computer vision domain. The SC aims to describe shapes that helps measure/check shape similarity and the recover/reconstruct point of correspondences [33]. In their explanation, the first step is to pick n points (shape contour). For each point p_i, there are $n - 1$ vectors that can be produced by connecting p_i to all remaining points on the contour. As a consequence, the set of all these feature vectors can enrich shape description at every specific point. Since it takes all points and their corresponding description (vectors), the overall feature matrix is found to be far too detailed and, therefore, is highly discriminant. We found that it is robust to small perturbations. At the same time, it does not guarantee scale-invariance.

4.1.2 Region-Based Shape Analysis

As compared to contour-based descriptors, region-based descriptors take into account all pattern pixels. In other words, a complete information can be preserved.

- Moment theory: More often, moment theory that includes methods, such as geometric, Legendre, Zernike, and pseudo-Zernike moments [34–42]. Moments with invariant property are important to pattern recognition, and we found that it was introduced in 1960 [43]. Further, since we have more studies in the domain, we must be aware of their fair comparison [34, 37]. The primary idea behind comparative studies is to see whether we have improved invariance properties, such as rotation, translation, and scaling [35, 42, 44–46]. At the same time, like before, reducing the processing time of the *Zernike* moments is the must [47]. In a similar fashion, the proposed orthogonal moments (computed from either binary or grayscale image [40] can help reconstruct the image [41].
- Fourier descriptor: Not to be confused, FD can also be used as a region-based shape descriptor. In 2002, Zhang and Lu [48] proposed a region-based generic Fourier descriptor (GFD). The FGD can overcome the limitations of contour-based FD. It uses the 2D Fourier transform (FT) on a polar-raster sampled shape. Polar representation can help avoid the rotation issue (in the Fourier spectra). Further, the GFD outperforms common contour-based (classical Fourier and curvature approaches)

and region-based (Zernike moments) shape descriptors [48].
However, high processing time is the major demerit of these approaches. We also found that possibility of optimization and complexity reduction has been reported [47, 49]. In brief, in the following, let us summarize region-based descriptors:

(a) Normalization process (representing global shape of the pattern by a single vector, for instance), is used to satisfy common geometric invariance properties. For example, normalization process includes centroid computation, re-sampling, and re-quantization can help.
(b) A single feature vector may not always capture complete shape information about the pattern. In other words, the descriptor is less discriminant. Its discrimination power and robustness can be extended by selecting an optimal set of features, including the suitable classifiers as well as their possible combination [50–53]. Detailed idea of how can classifiers can be combined is not the scope of the book. But, as an example, in the field of graphical symbol recognition, descriptors were combined with different classifiers. The whole process helps boost the recognition performance [54].

On the whole, it is not straightforward to categorize strictly state-of-the-art literature into contour- and region-based descriptors, since their mathematical/theoretical background looks similar. We found that their choice/selection relies on the need and the nature of the studied sample, i.e., application dependent.

4.2 Graphics Recognition

In statistical pattern recognition, common geometric properties are centroids, axes of inertia, circularity, area, line intersections, holes and projection profiles, geometric moments, and image transformations. This holds true for graphical symbol recognition. Following the statistical approaches, let us summarize graphics recognition in detail.

- As discussed earlier, shape descriptors consider the pattern's global appearance. Let us have more detailed explanation in [55].
  However, a set of few geometric properties/features, such as the degree of ellipticity i.e., the ratio of major to minor axis [40], the maximum orientation axis [56] and fuzzy topological attributes [57–59] may not be suitable for complex graphical symbol recognition. To separate isolated symbols with distinct shapes [3], the aforementioned properties may work. In [60], the authors derived the angular radial transform (ART). For graphics recognition, it is considered as a powerful method helps produce optimal performance [61]. It is important to note that the concept was initially tested for the MPEG problem.

- Global shape representation is popular due to its implementation simplicity. Further, it does not require additional preprocessing and segmentation as compared to local descriptor in the computer vision domain. As an example, the GFD concept was clearly reported in [8]. Besides, in [62], global shape descriptors were tested for graphical symbol recognition, where processing time was also included as an important factor. In [63], authors observed that results (using multi-oriented and multi-scale character dataset) are generally better than both Zernike moments and circular features.

  The Radon transform (RT) is found to be another popular descriptor in the domain. It can globally describe a pattern by taking all possible projections into account [64]. Note that it is derived from the Trace transform [65]—a widely used pattern representation method. In many applications, to represent pattern, the RT was combined with the distance transform or with distance between the regions-of-interest in the angular scans [66–68]. Their reported results were encouraging. Let us have more detailed explanation. In [68], the feature vector representing shape distribution is computed after normalizing all possible projection profiles i.e., $(0, \pi[$. As a consequence, it eases (speeds up) matching process . However, it does not carry complete shape information about the studied pattern. Since the authors observed that a single feature vector cannot preserve complete information, they introduced histograms (profiles) of the RT instead of compressing them into a single vector [69]. In this context, the authors assumed that the studied patterns are of exactly the same size. At this point, note that the RTs are essentially histograms or profiles. In contrast, in [5, 70], since pattern sizes can be varied, dynamic time warping (DTW) is used to match corresponding histograms for all projections. The use of the DTW absorbs varying histogram sizes due to the change in image size. The primary advantage of the use of the DTW is that it can avoid feature matrix compression into a single vector. Compression process helps miss information. In their approach, we still need to understand that the DTW is a commonly used technique that was used to compute distances between two profiles [71].
- If we consider complex and composite graphical symbol, we are required to employ robust descriptors that can describe the pattern without significant loss of information. Further, it may not be appropriate to use exactly the same descriptor for multiple applications, since there are several issues like variability of appearance. Such a variation refers to rotation, partial occlusions, elastic deformations, and intra-class and inter-class differences.

  In [72], their shape descriptor aimed to deal with the changes in appearance. Following the idea of principal components, recognition process can be made rotation and reflection invariance. Within this framework, the blurred shape model descriptor (BSM) encodes the probability of appearance of each pixel that outlines the shape of the pattern.

  In [73], the authors introduced a concept that can learn models based on the shape variations. It relies on the active appearance model (AAM). The concept allows us to integrate shape deformations. They have considered the BSM descriptor that can handle basic shape and appearance. This clearly tells us that it overcomes the limitations that were in original BSM. Their results were interesting.

- In [74], a new descriptor was proposed. It creates a histogram for each pixel by taking remaining pixels into account. A feature vector is built by taking all possible histograms that are statistically combined. The authors found that the descriptor is appropriate for graphical symbol recognition (technical drawings). Based on its working principle, its shape is found to be similar to the skeleton. As a result, it may not be precise for patterns, where one needs to differentiate topological properties. It suffers from high computational cost. In [75], Kullback–Leibler divergence technique is used to assess similarity between graphical symbols, where symbols are represented as 2D kernel densities. In their results, it produces high accuracy in comparison to other state-of-the-art methods in the domain. On the other hand, these methods [74, 75] were not tested on complex and composite symbols. As a result, we cannot guarantee its usefulness on complex and composite graphical symbol recognition.
- In [76], the author introduced a connection between the sparse representation concept and the use of vocabulary. Using local descriptors, visual vocabulary is built in terms of dictionary. After that, following its sparse representation, they construct a vector model for every symbol and adapt the tdf-if approach. For both isolated, and deformed and degraded symbols, the method is found to be encouraging, since it outperforms the state-of-the-art shape descriptors, such as Zernike moments [44] and R-signature [68]. However, the results were not reported on complex and composite symbols [77] (see Fig. 4.3) and other symbol spotting problems [78, 79].
- For shape analysis and graphical symbol recognition, a few works presented earlier can help understand more [3, 4, 80]. In these articles, we observe the usefulness and/or importance of the shape descriptors for document analysis and a collection of techniques that are used for graphical symbol recognition. Most of them (techniques) are suitable for isolated line symbols, regardless of their source: machine printed and handwritten. Moreover, for degradation (due to noise), occlusions and all sort of variants can be handled by statistical approaches. In particular, in [3], shape analysis for symbol representation refers to simple isolated $2D$ binary shapes. Few test samples are shown in Figs. 4.1 (machine printed) and 4.2 (handwritten). In both cases, degradation, noise (at different levels) can be seen, and therefore, complexity has been extended. Further, missing parts (lines, see Fig. 4.2) could make the problem more difficult.

But these statistical approaches may not suitable for composite symbols that are connected texts and other graphical elements, for instance [4, 77–79, 83]. In statistical approaches, global signal-based descriptors [8, 33, 44, 48, 68, 84] cannot be precise, since they do not preserve small changes within the symbol. As a result, like we have discussed earlier, it may not work for complex and composite symbols (see Fig. 4.3). Besides, Fig 4.4 can help understand the real-world example and its difficulty. More specifically, considering the way how global signal-based descriptors are formed, deformation can lead to centroid change that produces errors in the ring projection [84] and occlusion can lead to the change in tangents along the boundary (shape context [33], for instance). Besides, we have observed that integrating

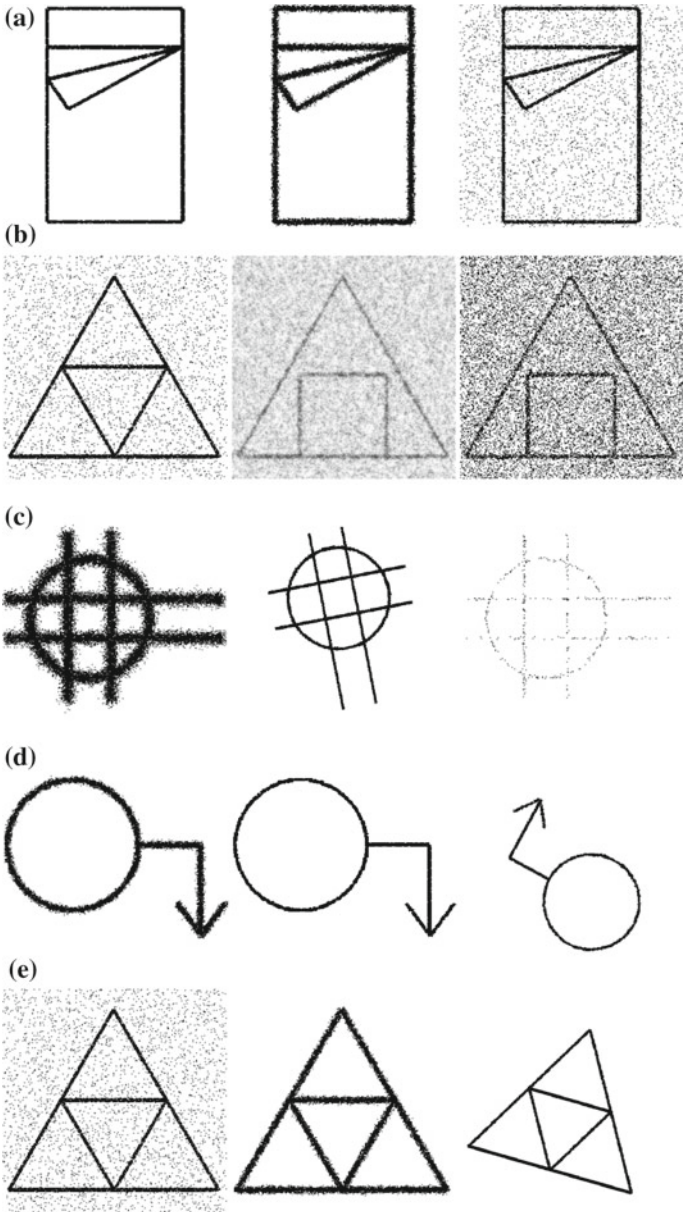


Fig. 4.1 Few test sample images from GREC'05 [81] (source: DAE, http://dae.cse.lehigh.edu/DAE/)

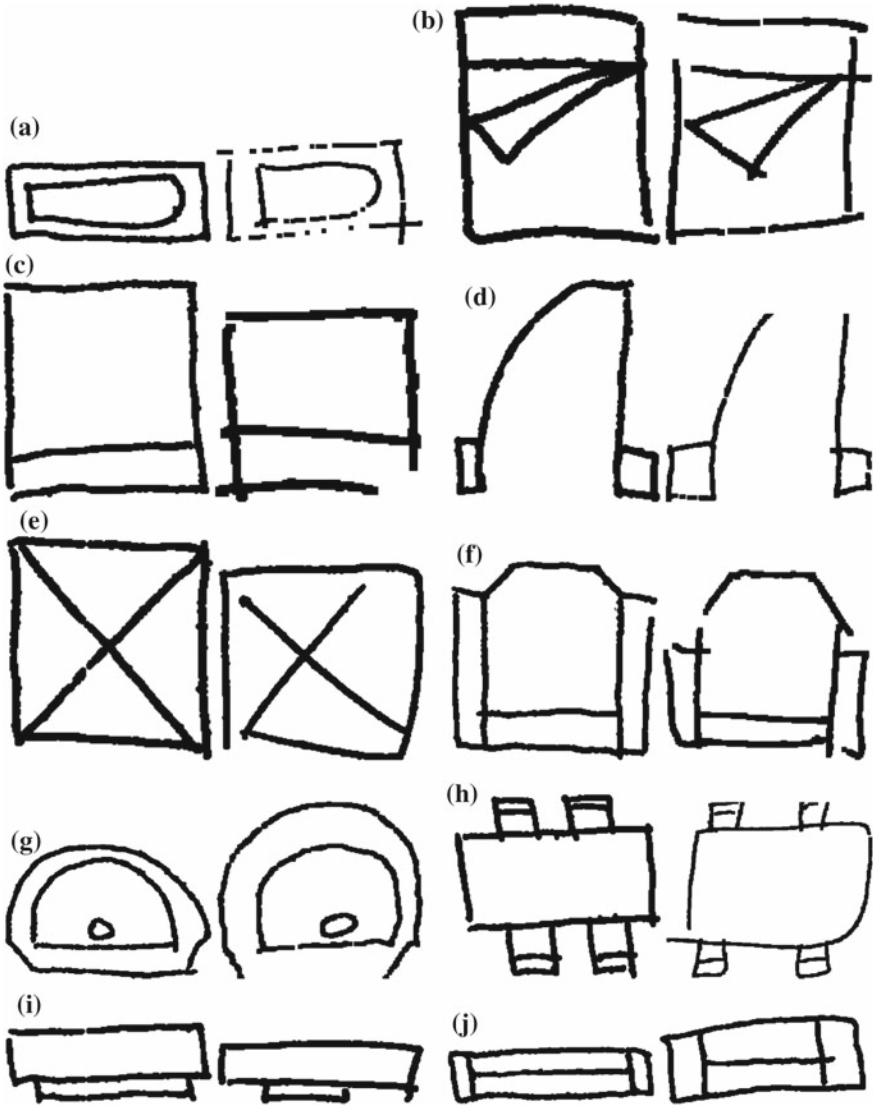


Fig. 4.2 2 hand-drawn samples from 10 different known classes (reported in [71, 82])

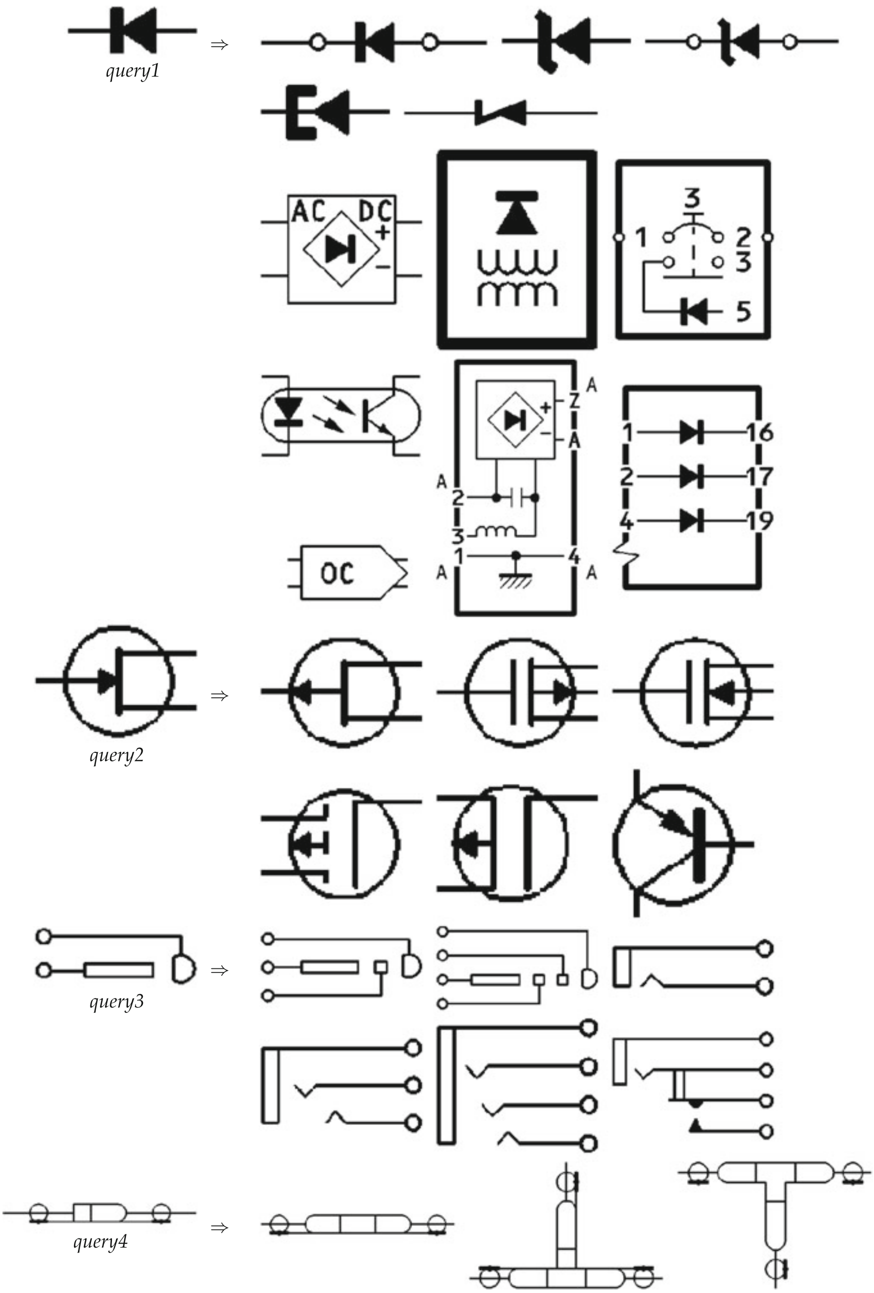


Fig. 4.3 Few sample images (queries and the corresponding symbols) from the FRESH dataset [77]). For every query symbol: *query1* to *query4*, few relevant symbols are shown (regardless their order) based on their similarity (experts). It has both linear as well as symbols in the composite form that includes texts

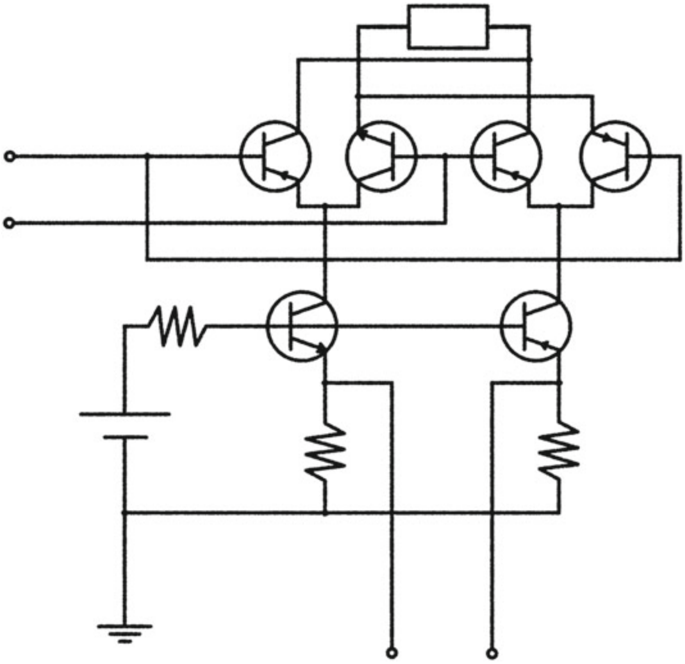


Fig. 4.4 A sample showing symbols have to be identified from the circuit diagram [91] (source: DAE, http://dae.cse.lehigh.edu/DAE/)

descriptors [85–87] and several machine learning classifiers [88] can help boost their performance because off-the-shelf methods are designed for isolated line symbols [54]. As an example, in [85], we have found that several features (such as compactness, ellipticity, angular features, and GFD) are combined. Similarly, in [89, 90], a set of statistical features can be used to partition line drawings into shapes based on how meaningful the parts of the symbols are. The proposed technique provided an accurate and consistent regions-of-interest or meaningful parts detection. Regions-of-interest refers to the conversion of a complete line drawing into a set of isolated shapes. But the downside of their technique is it has high computational cost.

On the whole, except for a few algorithms that are especially developed for specific tasks, statistical approaches are simple to compute with low computational cost as compared to other techniques, such as graph-based pattern representation and recognition that comes under structural approaches. Statistical signatures are simple to compute that does not mean that it has not discrimination power. Discrimination power and robustness strongly depend on how we select the set of features.

4.3 Experiments

Since the difficulties, in graphics recognition, are observed in complex and composite graphical symbol recognition. In what follows, let us have a quick overview (experimental results) of how far shape descriptors (global) can handle. In our discussions (see previous section), we have discussed that within the framework of shape analysis in pattern recognition—graphics recognition, in particular—the Radon transform (RT) [64]-based techniques are widely adopted because [5]

1. It is a rich transform, where each pattern's point lies on a set of lines in the spatial domain and contributes a curve to the Radon image;
2. It is a lossless transform and pattern reconstruction is possible; and
3. It is possible to make rotation, scaling, and rotation transformations invariant.

A summary of how the concept of the DTW-Radon [5] has been derived/motivated can be explained as follows. The RT yields $\mathscr{R}$-signature [68] by taking an integral function and the discrete Fourier transform on the radial and angular slices of the Radon image. In a similar fashion, the RT yields ϕ-signature [92], which is computed by applying an integral function on the angular slices of the Radon image so that rotation invariance property can be maintained. A basic normalization process can help provide translation and scaling invariant. These signatures are simple to compute. However, they are less discriminant since no complete information can be preserved. It is primarily because of the compression process at the time of transforming the Radon image into 1D signature. Besides, the normalization process does not help preserve fine details and, therefore, the signatures computed from noisy images may not be accurate enough to make a difference. We have observed that the $\mathscr{R}$-signature, without a surprise, has been adopted by multiple applications, such as graphical symbol recognition [88] and activity recognition [93, 94]. After that, the generalized version of $\mathscr{R}$-transform was reported in the year 2012 [95], where more insights to theoretical geometric interpretations were discussed. Specifically, it addresses exponent values variation at the time of using integral function, instead of just using numeric value, i.e., 2 as reported/studied in [68]. The generalized version opens doors since one needs to tune that value since the optimal value varies from one dataset to another, i.e., application dependent. This means that the exact same value cannot be used. The results vary as soon as exponent value changes. The effect can be more pronounced in case we use noisy images. As an example, high exponent value may cause high variation in the $\mathscr{R}$-transform [95]. As a result, the signature is different from the ideal analytical values. This means that the $\mathscr{R}$-transform with different exponent values (except 1) due to noise may not be suitable for recognition. In addition, their techniques use the compression of the Radon image [68, 92], it has similar limitations as discussed before. The similar statement holds true for other work [96].

Instead of relying on the $1D$ signatures, use of the histograms of the Radon transform (HRT)in [69] could be a better idea. In contrast to previous work [68], the HRT is more efficient as it produces $2D$ histogram. However, such a set of histograms is not invariant to rotation. As a consequence, these histograms need to be corrected by taking image rotation angle so that corresponding histograms can be matched for similarity checking between two different patterns. For these histograms, image size is an important factor. This means that to make histograms of equal sizes, images are scaled into a fixed size window. At the same time, lossless resizing/scaling techniques has to be applied. If not, shape information of degraded patterns can be skewed because of the wrong aspect ratio [97]. Therefore, in many applications, heavy computational time could be a better idea, where it has to discriminate several shape deformations, such as degradations and distortions. In [98], authors employed phase-only-correlation (POC) algorithm that helps satisfy rotation-invariant property, where logarithmic of HRT (LHRT) was used to normalize the histograms. With this concept, POC may provide several peaks (similar magnitudes) in case periodic images (chess-board, for instance) are considered. In brief, such approaches may not be suitable for deformed patterns (degradations and distortions, occlusions, and nonuniform scaling).

We note that the RT is a set of histograms or features that can be parametrized. In order to fully exploit information of the Radon image, resizing an image may not be a wise idea nor does compressing the Radon histogram work. Therefore, the use of dynamic programming can be an interesting alternative because of the following reasons:

(a) Dynamic programming (Dynamic time warping (DTW)) can absorb varying histogram sizes at the time of matching.
(b) In contrast to previous works [70, 99, 100], the optimal selection of number of histograms is another important issue as it can reduce computational cost.
(c) To reduce the number of DTW matchings, a priori knowledge about the pattern orientation can be used to make one-to-one histogram matching.
(d) With this idea of using DTW, broader perspectives can be established by taking several different problems, such as different levels of noise, distortion, and occlusion.

In [5], more information can be found.

4.3.1 DTW-Radon: How Does It Work?

4.3.1.1 Shape Representation

Radon transform (RT) [64] is a set of projections of the shape of the pattern at different angles. In other words, for an image pattern $P(x, y)$ and for a given set of angles, the RT can be taken as the projection of all nonzero points that eventually produces a matrix.

Mathematically, the integral of P over a line $\mathscr{L}(\rho, \theta)$ defined by $\rho = x\cos\theta + y\sin\theta$ can be expressed as

$$R(\rho,\theta)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}P(x,y)\delta(x\cos\theta+y\sin\theta-\rho)dxdy, \tag{4.1}$$

where $\delta(.)$ is the Dirac function,

$$\delta(x)=\begin{cases}0 \text{ if } x\neq 0\\ 1 \text{ otherwise,}\end{cases}$$

with $\theta\in[0,\pi[$ and $\rho\in]-\infty,\infty[$. For the RT, $\mathscr{L}_i$ be in normal form (ρ_i,θ_i). For all θ_i, the RT now can be described as the length of intersections of all lines $\mathscr{L}_i$. Note that the range of ρ i.e., $-\rho_{min}<\rho\leq\rho_{max}$ is entirely based on the size of pattern. A complete illustration is provided in Fig. 4.5.

Can we make the RT transformation invariant (affine transformation: translation, scaling, and rotation) [101]?

1. *Translation*: Consider image centroid (x_c,y_c) such that translation vector is $\mathbf{u}=(x_c,y_c)$ and the Radon transform is $R(\rho-x_c\cos\theta-y_c\sin\theta,\theta)$. This results the shift of its transform in ρ by a distance equal to the projection of translation vector of the line $\mathscr{L}$. This helps the RT to be translation invariant.
2. *Scaling*: Normalization can help, i.e., the profiles/histograms can be normalized into [0, 1] at every projecting angle.
3. *Rotation*: To make corresponding histogram/profile matching efficient, a priori knowledge about the orientation angle can be computed as [102],

$$\alpha=arg\left[\min_{\theta}\frac{d^2\sigma_\theta^2}{d\theta^2}\right], \tag{4.2}$$

where variance and mean of projection at θ are

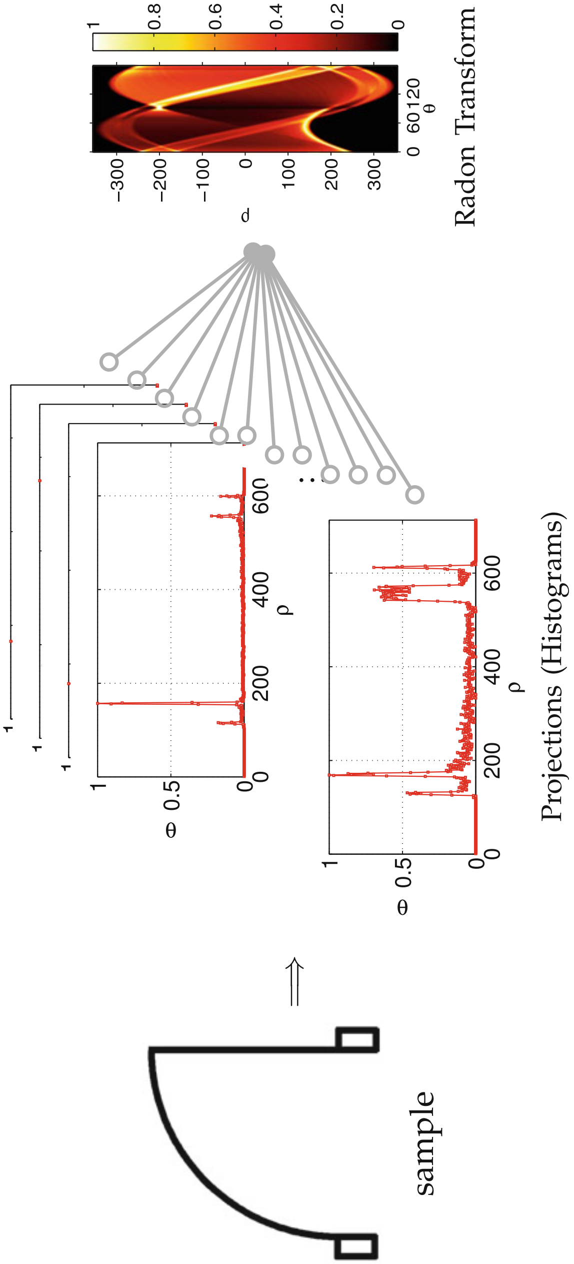
$$\sigma_\theta^2=\frac{1}{P}\sum_{\rho}(R(\rho,\theta)-\mu_\theta)^2$$

$$\mu_\theta=\frac{1}{N}\sum_{\rho}R(\rho,\theta)$$

and N is the number of samples. If angle of rotation is α, then $R^{\alpha}(\rho,\theta)=R(\rho,\theta+\alpha)$. This means that, we are required to make a circular shift of the profiles/histograms.

In [102], the RT is tested to detect linear trends in images. In other words, a pattern's principal/global orientation can be estimated by considering presence of the pixels in that particular direction/angle. This means that the RT profile along this direction usually has larger variations. Mathematically, the variance of the projection is locally maximum. In case of multiple local maxima, the second derivative of the variance would provide a unique solution. We have also noted that the derivative removes the low-frequency components of the variance.

Fig. 4.5 The Radon features for all possible projections over $[0, \pi[$ and the complete Radon transform i.e., a collection of all the Radon histograms (features)



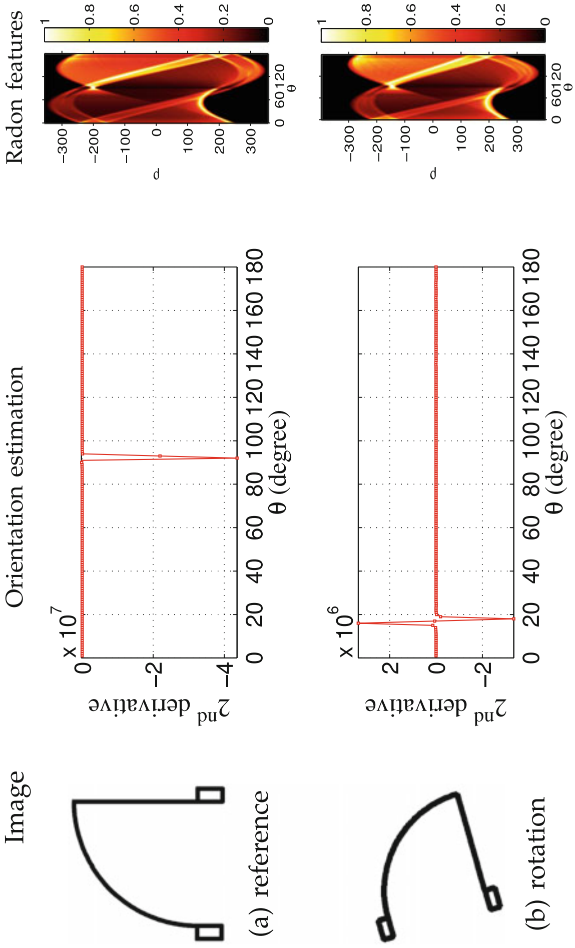


Fig. 4.6 Images, their corresponding orientation estimation and the Radon features, where GREC03 test samples are used [103]. Orientation angles α are estimated as follows: 90° for reference and 17° for rotation sample

For better understanding, in Fig. 4.6, the Radon features for reference and rotation (a known class of graphical symbol) are shown. In this example, estimated the orientation angles and the Radon histograms (features) from their corresponding images are shown. Note that the idea behind the use of rotation correction before corresponding profile matching can reduce recognition time.

Let us repeat, the RT is a set of parametrized profiles/histograms that are in the range: $[0, \pi[$, where π is not included. Formally, a complete set $R(\rho, b)$ of the Radon features can be expressed as

$$\mathscr{F} = \{\mathbf{F}_b\}_{b=1,\ldots,B}, \tag{4.3}$$

where the total number bins B can be formulated as

$$B = \frac{180}{\Theta} = \begin{cases} 180 & \text{when } \Theta = 1^\circ \\ 90 & \text{when } \Theta = 2^\circ \\ 60 & \text{when } \Theta = 3^\circ \\ 36 & \text{when } \Theta = 5^\circ \\ \ldots & \text{so on,} \end{cases}$$

with Θ as the projection angle-range. As said before, to avoid loss of information, compressing a matrix into a 1D signature is not the choice. In general, a single Radon feature at bin b is $\mathbf{F}_b$ is the set of histograms. In case when $\Theta = 1^\circ$, each projecting angle can represent a bin. Altogether, we have 180 bins. If $\Theta = 5^\circ$, all histograms within the range are averaged to produce a single feature and there are 36 bins in total. To perform experiments, different numbers of bins can be used. Based on the value of B, there will be different outputs. In other words, performance function (f) is parametrized by B, i.e., $f(B)$. For any particular test, $\arg\max_B f(B)$ can help decide the optimal number of bins.

4.3.1.2 Shape Matching

As explained before, we have set of features $\mathscr{F}$ in a specified number of bins B for any pattern P. Given two patterns: query P^q and database P^d, matching can be made by computing distance between two corresponding features (*cf.* Eq. (4.3)) from the complete sets of $\mathscr{F}^q$ and $\mathscr{F}^d$. Since the RT produces different ρ sizes in accordance with the pattern size, let us employ DTW algorithm [104]. In what follows, let us discuss matching process first and then matching/(dis)similarity score between two patterns.

Consider we have two feature vectors $\mathbf{X} = \{\mathbf{x}_k\}_{k=1,\ldots,K}$ and $\mathbf{Y} = \{\mathbf{y}_l\}_{l=1,\ldots,L}$ of size K and L, respectively, and $K \neq L$. To provide the optimal alignment between both vectors potentially having different lengths, we first construct a matrix M of size $K \times L$. We then, for each element in matrix M, compute local distance $\delta(k, l)$ between the elements e_k and e_l. With this procedure, we compute $D(k, l)$, which is defined as the global distance up to (k, l),

$$D(k,l) = \min \begin{bmatrix} D(k-1, l-1), \\ D(k-1, l), \\ D(k, l-1) \end{bmatrix} + \delta(k,l) \tag{4.4}$$

with an initial condition $D(1, 1) = \delta(1, 1)$ such that it allows warping path, which follows diagonally from $(1, 1)$ to (K, L), i.e., start to end. The basic idea is to find the path from which the least cost is produced. This means that the warping path provides the difference cost between two features. Formally, the warping path is, $\mathscr{W} = \{w_t\}_{t=1\ldots T}$, where $\max(k, l) \leq T < k + l - 1$ and tth element of $\mathscr{W}$ is $w(k, l)_t \in [1 : K] \times [1 : L]$ for $t \in [1 : T]$. The optimized warping path $\mathscr{W}$ satisfies the following three conditions: (c1) boundary condition, (c2) monotonicity condition, and (c3) continuity condition:

c1. $w_1 = (1, 1)$ and $w_T = (K, L)$;
c2. $k_1 \leq k_2 \leq \cdots \leq k_K$ and $l_1 \leq l_2 \leq \cdots \leq l_L$; and
c3. $w_{t+1} - w_t \in \{(1, 1)(0, 1), (1, 0)\}$ for $t \in [1 : T - 1]$.

The condition, **c1** conveys that the path starts from $(1, 1)$ to (K, L), aligning all elements to each other. Another condition, **c2** helps path advance one step at a time. The final condition, **c3** limits steps in the warping path to adjacent cells. We also observe that **c3** implies **c2**. With all three different conditions, let us formally express the global distance between **X** and **Y** as,

$$\text{Dist.}\,(\mathbf{X}, \mathbf{Y}) = \frac{D(K, L)}{T}. \tag{4.5}$$

The last element of the $K \times L$ matrix gives the DTW-distance between **X** and **Y**, which is normalized by T. T the number of discrete warping steps along the diagonal DTW-matrix that goes diagonally from beginning to end of the matrix. Backtracking procedure helps find the the minimum cost index pairs (k, l) along the diagonal, starting from (K, L) by using DP:

$$w_{t-1} = \begin{cases} (1, l-1) & \text{if } k = 1 \\ (k-1, 1) & \text{if } l = 1 \\ \text{argmin} \left\{ \begin{matrix} D(k-1, l-1), \\ D(k-1, l), \\ D(k, l-1) \end{matrix} \right\} & \text{otherwise.} \end{cases}$$

In this implementation, we found that the lexicographically smallest pair has to be considered in case the "argmin" is not unique. The whole process is shown in Fig. 4.7, where two nonlinear feature vectors with different sizes are used.

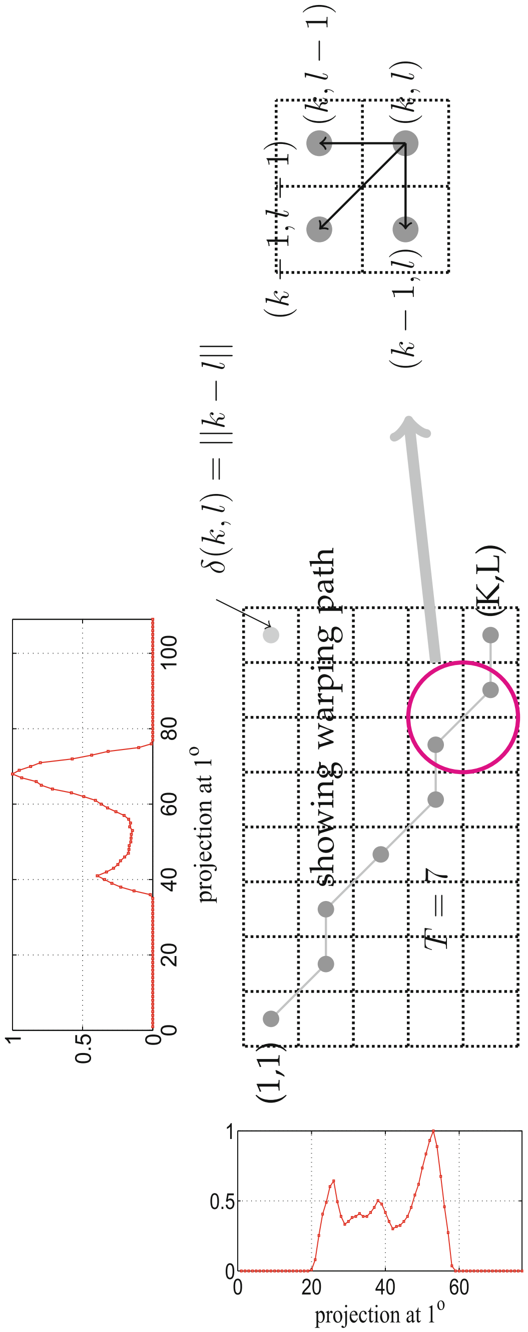
4.3.2 Results and Comparison

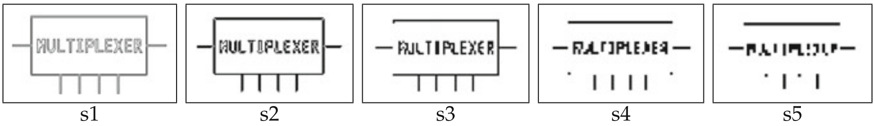
With this idea of shape representation and matching, in Fig. 4.8, we observe that the concept is robust to shape distortion and deformation (at different levels), where graphical symbol is considered. Having such encouraging performance, let us extend the test results on three different datasets:

(a) GREC [106],
(b) CVC [71] and
(c) FRESH [105].

While taking DTW-Radon into account, for several different number of bins B, it would be interesting to check how well other well-known shape descriptors can perform. For the test, the following shape descriptors are used:

Fig. 4.7 DTW algorithm





	s1	s2	s3	s4	s5
s1	0.00	1.50	2.60	4.40	5.50
s2		0.00	2.40	3.90	4.90
s3			0.00	3.20	3.60
s4				0.00	2.70
s5					0.00

(a) with α.

	s1	s2	s3	s4	s5
s1	0.00	1.20	2.50	4.30	6.10
s2		0.00	2.40	3.40	4.80
s3			0.00	3.10	3.50
s4				0.00	3.00
s5					0.00

(b) without α.

Fig. 4.8 Matching scores (in 10^{-3}) between distortion as well as deformed symbols (s1–s4), where test images are taken from FRESH dataset [105]

(a) $\mathscr{R}$-signature [68],
(b) GFD [48],
(c) SC [33] and
(d) ZM [44].

For all shape descriptors, selecting the parameters that are suitable for the test is a crucial factor. For the RT, projecting angle range is $[0, \pi[$, where all possible profiles are used. In case of GFD, both parameters radial $(4 : 12)$ and angular frequency $(6 : 20)$ parameters are tuned to select the best pairs, since the best pairs can be varied from one dataset to another. For SC, 100 sample points used as reported in [33]. In case of ZM, 36 *zernike* functions of order less than or equal to 7 are used.

In what follows, results are provided.

(a) *GREC dataset*

(i) *Dataset description.*
In this dataset[1] [103], different categories, such as ideal, rotation, scaling, distortion, and degradation are considered. Few samples are shown in Fig. 4.1.
In this dataset, 50 different models are categorized into 3 sets: set 1 (5 model symbols), set 2 (20 model symbols) and set 3 (50 model symbols). Each model symbol has 5 test images in every category except the ideal one. In their dataset, since vectorial distortion works only for symbols with straight lines (not arcs), 15 model symbols are vectorially distorted. Further, 9 different models of degradation are used.

(ii) *Results and analysis.*
For evaluation, each test image is matched with the model symbols to get the closest model. Since dataset is fully labelled and has balanced test images, as an evaluation metric, recognition can be applied (see Sect. 3.2 in Chap. 3).

[1] International symbol recognition contest, 2003.

Table 4.1 GREC dataset: average recognition rates (in %) for all data categories

Images set	DTW-Radon (with different B)						
	180	90	60	36	18	09	02
Ideal	100	100	100	100	100	100	100
Rotate	97	94	88	73	82	77	71
Scale	100	100	100	100	84	74	57
Rotate + scale	98	97	94	82	79	73	62
Distort	100	100	100	100	85	72	47
Degrade	99	98	95	84	67	47	35

Table 4.2 GREC dataset: comparison using the average recognition rates (in %) for all possible data categories

Images set	GFD	ZM	SC	$\mathscr{R}$-sign	DTW-Radon ($B = 180$)
Ideal	100	100	100	100	100
Rotate	98	94	97	94	97
Scale	99	98	99	96	100
Rotate + scale	98	93	98	92	98
Distort	100	94	100	92	100
Degrade	91	79	78	76	99

Test results for all types of aforementioned categories of datasets are shown in Tables 4.1 and 4.2. In Table 4.1, one can check how number of bins (DTW-Radon) affect the recognition performance, and Table 4.2 provides comparative study among other global shape descriptors. It is composed of results separately from ideal, rotation, scaling, combination of rotation and scaling, distortion and degradation categories. Following Tables 4.1 and 4.2, the following observations can be found.

- For ideal test images, one cannot judge the differences between shape descriptors. In DTW-Radon, it is important to notice that we have 100% recognition rates from all provided number of bins. For rotated images, GFD performs better, proving a marginal difference with SC and DTW-Radon for 180 bins. For scaled images, DTW-Radon outperforms all, where $B = 180$, 90, 60 and 36 provide 100% recognition rates. For other test images (rotation plus scale), DTW-Radon ($B = 180$), GFD and SC produce similar results.

  In brief, we cannot conclude that “the best” performer (shape descriptor). Besides, one cannot judge the superiority of the methods. Only execution time comparison would be an alternative.

- $\mathscr{R}$-signature does not provide satisfactory results on vectorial distorted test symbols. This does not hold true for other descriptors. At the same time, DTW-Radon (36 bins) can be compared with "the best" performer from the state of the art. In case of binary degradations, there exist notable differences between DTW-Radon and GFD.

(b) *CVC dataset*

(i) *Dataset description.*
This dataset comes from [71], which is of size 10×300 sample images. This means that there are 10 different known classes of hand-drawn architectural symbols with 300 samples per class. In Fig. 4.2, few samples are shown. The dataset has different problems, such as distortions, gaps, overlapping as well as missing parts within the shapes.

(ii) *Results and analysis.*
For evaluation, each test sample is matched with known classes and number of correct matches is reported over the requested list. In our dataset, there are 300 samples per class, the size of the requested list can be 300. This means that we can retrieve all similar images from every class including itself. In Tables 4.3 and 4.4, the average retrieval rates for all requested shortlists (e.g., top-20, top-40, and so on, i.e., increasing steps of 20) are provided. In Table 4.3, one can check how the number of bins affects the recognition performance, while Table 4.4 compares DTW-Radon results with other global shape descriptors.
In contrast to GREC dataset, SC provides the best performance from the list. However, it cannot outperform DTW-Radon.
In our retrieval list, up to top-60, we do not see differences between the methods. After top-60, one can make a difference. This means that if we are looking for retrieval stability, we should not stop at top-60. In the latter framework, DTW-Radon outperforms SC by more than 16%. SC lags GFD by approximately 9%. Further, ZM follows $\mathscr{R}$-signature.
In brief, in contrast to benchmarking descriptors, recognition rates from DTW-Radon for $B = 180, 90, 60$ and 36 provide better results.

(c) *FRESH dataset*

(i) *Dataset description.*
Let us consider a real-world problem (industrial project). In our dataset, we are required to identify a set of different known symbols in aircraft electrical wiring diagrams [105] (see Fig. 4.3). It has been clearly mentioned before that symbols may either be very similar in shape. They can be having slight changes in the shape. Further, they may also be composed of other known and graphical cues/elements. They do not necessary be connected. In addition, texts are appeared. It is composed of roughly 500 different known symbols. This dataset has no absolute ground-truth, and human validation was required, where possible subjective bias was avoided [77, 107–110].

Table 4.3 CVC dataset: average retrieval rate (in %)

Requested list	DTW-Radon (with different B)						
	180	90	60	36	18	09	02
Top-20	99	99	99	98	94	92	83
Top-40	99	99	98	97	92	88	79
Top-60	97	97	97	96	90	86	67
Top-80	97	97	97	94	88	81	64
Top-100	97	97	96	94	84	80	61
top-120	97	96	96	93	80	74	60
Top-140	95	95	95	92	76	67	58
Top-160	95	95	94	91	73	64	57
Top-180	93	93	94	89	67	61	54
Top-200	93	92	91	86	62	56	53
Top-220	93	92	91	86	59	53	50
Top-240	92	90	88	84	55	51	49
top-260	91	89	86	83	52	44	46
Top-280	88	87	85	81	48	42	45
top-300	86	86	84	78	47	39	44

Six volunteers were used to manually select what they consider as “similar” symbols, for all queries executed in this section. While selection, they did not provide any ranking order nor degree of visual resemblance (shape similarity).

(ii) *Results and analysis.*

Since the aim is not only limited to recognizing symbols but also extends to ranking them. Note that the number of ground-truths varied from one query to another. As traditional precision and recall cannot be used for imbalanced dataset, retrieval efficiency [111] can be used as explained in Sect. 3.2 of Chap. 3.

In Tables 4.5 and 4.6, retrieval efficiencies are provided for $\mathscr{K}$ ranging from 1 to 10. From the set of benchmarking shape descriptors, GFD performs better. GFD is followed by SC and then ZM. DTW-Radon (with $B = 180, 90, 60$) outperforms all of them. The minimum score from DTW-Radon with $B = 36$ can also be compared.

Overall, considering global shape descriptors, DTW-Radon is found to be consistent. However, it does not clearly tell it can always outperform others. The idea behind a series of test results is that shape descriptors are application dependent (nature of the dataset).

Table 4.4 CVC dataset: comparison using the average retrieval rates (in %)

Requested list	GFD	ZM	SC	$\mathscr{R}$-sign.	DTW-Radon ($B = 180$)
Top-20	96	68	98	82	99
Top-40	93	62	95	75	99
Top-60	90	59	95	69	97
top-80	88	57	92	65	97
Top-100	85	55	91	62	97
Top-120	83	54	88	59	97
Top-140	81	52	87	56	95
Top-160	78	50	85	54	95
Top-180	76	50	83	51	93
top-200	73	48	81	49	93
Top-220	71	44	78	48	93
Top-240	68	41	78	46	92
Top-260	66	39	75	45	91
Top-280	63	37	73	43	88
Top-300	61	36	70	42	86

Table 4.5 FRESH dataset: retrieval efficiency (in %) over 30 queries

Requested list	$\mathscr{D}$-Radon (with different B)						
	180	90	60	36	18	09	02
Top-2	92	92	91	87	78	77	75
Top-4	83	82	81	76	66	62	60
Top-6	77	75	74	69	57	53	50
Top-8	76	75	73	64	49	45	45
Top-10	73	71	69	58	44	42	41

Table 4.6 FRESH dataset: comparison using the retrieval efficiencies (in %) over 30 queries

Requested list	GFD	ZM	SC	$\mathscr{R}$-sign	DTW-Radon ($B = 180$)
Top-2	91	88	87	84	92
Top-4	80	72	72	71	83
Top-6	74	65	63	60	77
Top-8	71	60	59	51	76
Top-10	69	56	54	49	73

4.4 Summary

In this chapter, statistical approaches have been covered for graphics recognition, where we have detailed several statistical techniques/algorithm including their usefulness. In addition to recognition and/or retrieval performance, time complexity issue has been addressed. We have observed that the primary advantage of such approaches is robust to noise, degradation, occlusions, and deformations. These are, however, not useful for complex and composite graphical symbols. For such a latter issue, structural approaches could be a better choice. In the next chapter, we will discuss about structural approaches for graphics recognition.

References

1. P.K. Ghosh, K. Deguchi, *Mathematics of Shape Description: A Morphological Approach to Image Processing and Computer Graphics* (Wiley, 2008)
2. J. Gauch, *Multiresolution Image Shape Description* (Springer, Berlin, 1992)
3. L.P. Cordella, M. Vento, Symbol and shape recognition, iIn *Proceedings of 3rd International Workshop on Graphics Recognition, Jaipur (India)* (1999), pp. 179–186
4. L.P. Cordella, M. Vento, Symbol recognition in documents: a collection of techniques? Int. J. Doc. Anal. Recogn. **3**(2), 73–88 (2000)
5. K.C. Santosh, B. Lamiroy, L. Wendling, Dtw-radon-based shape descriptor for pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **27**(3), 1350008 (2013)
6. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, 2015), pp. 1–22
7. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi. Communications in Computer and Information. Science **709**, 3–15 (2017)
8. D. Zhang, G. Lu, Review of shape representation and description techniques. Pattern Recogn. **37**(1), 1–19 (2004)
9. K.C. Santosh, B. Lamiroy, L. Wendling, DTW for matching radon features: a pattern recognition and retrieval method, in *Advances Concepts for Intelligent Vision Systems (ACIVS)* (2011), pp. 249–260
10. H. Kauppinen, T. Seppänen, M. Pietikäinen, An experimental comparison of autoregressive and fourier-based descriptors in 2D shape classification. IEEE Trans. Pattern Anal. Mach. Intell. **17**(2), 201–207 (1995)
11. Y. Rui, A. She, T.S. Huang, A modified fourier descriptor for shape matching in mars, in *Image Databases and Multimedia Search*, ed. by S.K. Chang. Software Engineering and Knowledge Engineering, vol. 8 (World Scientific Publishing House in Singapore, 1998), pp. 165–180
12. C.H. Lin, New forms of shape invariants from elliptic fourier descriptors. Pattern Recogn. **20**, 535–545 (1987)
13. E. Persoon, K. Fu, Shape discrimination using fourier descriptors. IEEE Trans. Syst. Man Cybern. **7**(3), 170–179 (1977)
14. D. Zhang, L. Guojun, Study and evaluation of different fourier methods for image retrieval. Image Vis. Comput. **23**(1), 33–49 (2005)
15. A. El-ghazal, O. Basir, S. Belkasim, Farthest point distance: a new shape signature for fourier descriptors. Sig. Process. Image Commun. **24**(7), 572–586 (2009)
16. T. Taxt, J.B. Olafsdottir, M. Daehlen, Recognition of handwritten symbols. Pattern Recogn. **23**, 1155–1166 (1990)
17. M. Maes, Polygonal shape recognition using string-matching techniques. Pattern Recogn. **24**(5), 433–440 (1991)

18. E. Attalla, P. Siy, Robust shape similarity retrieval based on contour segmentation polygonal multiresolution and elastic matching. Pattern Recogn. **38**(12), 2229–2241 (2005)
19. R. Gerdes, R. Otterbach, R. Kammuler, Fast and robust recognition and localization of 2D objects. Mach. Vis. Appl. **8**, 365–374 (1995)
20. D.H. Ballard, Generalizing the hough transform to detect arbitary shapes. Pattern Recogn. **13**(2), 111–122 (1981)
21. P. Fränti, A. Mednonogov, V. Kyrki, H. Kälviäinen, Content-based matching of line-drawing images using the Hough transform. Int. J. Doc. Anal. Recogn. **3**(2), 117–124 (2000)
22. A.A. Kassim, T. Tan, K.H. Tan, A comparative study of efficient generalised Hough transform techniques. Image Vis. Comput. **17**, 737–748 (1999)
23. C.-P. Chau, W.-C. Siu, Adaptive dual-point hough transform for object recognition. Comput. Vis. Image Underst. **96**(1), 1–16 (2004)
24. M. Pelillo, K. Siddiqi, S.W. Zucker, Matching hierarchical structures using association graphs. IEEE Trans. Pattern Anal. Mach. Intell. **21**(11), 1105–1120 (1999)
25. F. Mokhtarian, S. Abbasi, Shape similarity retrieval under affine transforms. Pattern Recogn. **35**(1), 31–41 (2002)
26. C. Urdiales, A. Bandera, F. Sandoval, *Hernndez* (Pattern Recognition, Non-parametric planar shape representation based on adaptive curvature functions, 2002), pp. 43–53
27. T. Bernier, J.-A. Landry, A new method for representing and matching shapes of natural objects. Pattern Recogn. **36**(8), 1711–1723 (2003)
28. B.B. Kimia, A. Tannenbaum, S.W. Zucker, Shapes, shocks, and deformations i: the components of two-dimensional shape and the reaction-diffusion space. Int. J. Comput. Vis. **15**(3), 189–224 (1995)
29. S.C. Zhu, A.L. Yuille, FORMS: a flexible object recognition and modelling system. Int. J. Comput. Vis. **20**(3), 187–212 (1996)
30. D. Sharvit, J. Chan, H. Tek, B.B. Kimia, Symmetry-based indexing of image databases. J. Vis. Commun. Image Represent. **9**, 366–380 (1998)
31. K. Siddiqi, A. Shokoufandeh, S.J. Dickinson, S.W. Zucker, Shock graphs and shape matching. Int. J. Comput. Vis. **35**(1), 13–32 (1999)
32. T.B. Sebastian, P.N. Klein, B.B. Kimia, Recognition of shapes by editing their shock graphs. IEEE Trans. Pattern Anal. Mach. Intell. **26**(5), 550–571 (2004)
33. S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 509–522 (2002)
34. C.-H. Teh, R.T. Chin, On image analysis by the methods of moments. IEEE Trans. Pattern Anal. Mach. Intell. **10**(4), 496–513 (1988)
35. S.O. Belkasim, M. Shridar, M. Ahmadi, Pattern recognition with moment invariants: a comparative study and new results. Pattern Recogn. **24**, 1117–1138 (1991)
36. R.J. Prokop, A.P. Reeves, A survey of moment-based techniques for unoccluded object representation and recognition. CVGIP: Graph Models Image Process. **54**(5), 438–460 (1992)
37. R.R. Bailey, M. Srinath, Orthogonal moment features for use with parametric and non parametric classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **18**(4), 389–399 (1996)
38. R. Bamieh, R. de Figueiredo, A general moments invariants/attributed graph method for the three dimensional object recognition from a single image. IEEE J. Robot. Autom. **2**, 240–242 (1986)
39. Y. Chen, N.A. Langrana, A.K. Das, Perfecting vectorized mechanical drawings. Comput. Vis. Image Underst. **63**(2), 273–286 (1996)
40. R. Teague, Image analysis via the general theory of moments. J. Opt. Soc. Am. **70**(8), 920–930 (1979)
41. A. Khotanzad, Y.H. Hong, Invariant image recognition by zernike moments. IEEE Trans. Pattern Anal. Mach. Intell. **12**(5), 489–497 (1990)
42. S.X. Liao, M. Pawlak, On the accuracy of zernike moments for image analysis. IEEE Trans. Pattern Anal. Mach. Intell. **20**(12), 1358–1364 (1998)
43. M.K. Hu, Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory **8**, 179–187 (1962)

44. W.-Y. Kim, Y.-S. Kim, A region-based shape descriptor using zernike moments. Sig. Process. Image Commun. **16**(1–2), 95–102 (2000)
45. N.K. Kamila, S. Mahapatra, S. Nanda, Invariance image analysis using modified zernike moments. Pattern Recogn. Lett. **26**(6), 747–753 (2005)
46. C. Kan, M.D. Srinath, Invariant character recognition with zernike and orthogonal Fourier-Mellin moments. Pattern Recogn. **35**, 143–154 (2002)
47. C. Chong, P. Raveendran, R. Mukudan, A comparative analysis of algorithms for fast computation of zernike moments. Pattern Recogn. **36**, 731–742 (2003)
48. D. Zhang, L. Guojun, Shape-based image retrieval using generic fourier descriptor. Sig. Process. Image Commun. **17**(10), 825–848 (2002)
49. B.C. Lin, J. Shen, Fast computation of moment invariants. Pattern Recogn. **24**, 807–813 (1991)
50. J. Kittler, M. Hatef, R. Duin, J. Matas, On combining classifiers. IEEE Trans. Pattern Anal. Mach. Intell. **20**(3), 226–239 (1998)
51. M. Kudo, J. Sklansky, Comparison of algorithms that select features for pattern classifiers. Pattern Recogn. **33**(1), 25–41 (2000)
52. D. Ruta, B. Gabrys, An overview of classifier fusion methods. Comput. Info. Sys. **7**(1), 1–10 (2000)
53. R.O. Duda, P.E. Hart, D.G. Stork, *Pattern Classification*, 2nd edn. (Wiley Interscience Publications, New York, 2001)
54. K. Tombre, C. Ah-Soon, Ph. Dosch, A. Habed, G. Masini, Stable, robust and off-the-shelf methods for graphics recognition, in *Proceedings of the 14th International Conference on Pattern Recognition, Brisbane (Australia)* (1998), pp. 406–408
55. L. da Fontoura Costa, R.M. Cesar Junior, *Shape Analysis and Classification: Theory and Practice*. Book series on Image Processing (CRC Press, 2001)
56. R.M. Haralick, Performance evaluation of document image algorithms, in *Graphics Recognition—Recent Advances*, ed. by A.K. Chhabra, D. Dori. *Lecture Notes in Computer Science*, vol. 1941 (Springer, 2000), pp. 315–323
57. A. Rosenfeld, Image analysis: problems, progress and prospects. Pattern Recognit. **17**(1), 3–12 (1984)
58. A. Rosenfeld, A.C. Kak, *Digital Picture Processing*, vol. 2.8 (Academic Press, New York, 1982)
59. S.K. Pal, A. Rosenfeld, A fuzzy medial axis transformation based on fuzzy disks. Pattern Recogn. Lett. **12**(10), 585–590 (1991)
60. W.-Y. Kim, Y.-S. Kim, A new region-based shape descriptor, in *MPEG meeting, TR 15-01, Pisa (Italy)* (1999)
61. D. Zuwala, S. Tabbone, A method for symbol spotting in graphical documents, in *Document Analysis Systems VII: Proceedings of 7th International Workshop on Document Analysis Systems, Nelson (New Zealand)*, ed. by H. Bunke, A.L. Spitz. *Lecture Notes in Computer Science*, vol. 3872 (2006), pp. 518–528
62. S. Adam, J.M. Ogier, C. Cariou, R. Mullot, J. Labiche, J. Gardes, Symbol and character recognition: application to engineering drawings. Int. J. Doc. Anal. Recogn. **3**(2), 89–101 (2000)
63. N. Kita, Object Location Based on Concentric Circular Description **1**, 637–641 (1992)
64. S.R. Deans, *Applications of the Radon Transform* (Wiley Interscience Publications, New York, 1983)
65. A. Kadyrov, M. Petrou, The trace transform and its applications. IEEE Trans. Pattern Anal. Mach. Intell. **23**(8), 811–828 (2001)
66. S. Tabbone, L. Wendling, K. Tombre, Matching of graphical symbols in line-drawing images using angular signature information. Int. J. Doc. Anal. Recogn. **6**(2), 115–125 (2003)
67. S. Tabbone, L. Wendling, Binary shape normalization using the Radon transform, in *Proceedings of 11th International Conference on Discrete Geometry for Computer Imagery, Naples (Italy)*. *Lecture Notes in Computer Science*, vol. 2886 (2003), pp. 184–193
68. S. Tabbone, L. Wendling, J.-P. Salmon, A new shape descriptor defined on the radon transform. Comput. Vis. Image Underst. **102**(1), 42–51 (2006)

69. S. Tabbone, O. Ramos Terrades, S. Barrat, Histogram of radon transform. A useful descriptor for shape retrieval, in *Proceedings of the IAPR International Conference on Pattern Recognition* (2008), pp. 1–4
70. K.C. Santosh, B. Lamiroy, L. Wendling, DTW for matching radon features: a pattern recognition and retrieval method, in *Advances Concepts for Intelligent Vision Systems*, ed. by J. Blanc-Talon, R.P. Kleihorst, W. Philips, D.C. Popescu, P. Scheunders. *Lecture Notes in Computer Science*, vol. 6915 (Springer, 2011), pp. 249–260
71. A. Fornés, J. Lladós, G. Sánchez, D. Karatzas, Rotation invariant hand-drawn symbol recognition based on a dynamic time warping model. Int. J. Doc. Anal. Recogn. **13**(3), 229–241 (2010)
72. S. Escalera, A. Fornés, O. Pujol, J. Lladós, P. Radeva, Circular blurred shape model for multiclass symbol recognition. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **41**(2), 497–506 (2011)
73. J. Almazán, A. Fornés, E. Valveny, A non-rigid appearance model for shape description and recognition. Pattern Recogn. **45**(9), 3105–3113 (2012)
74. S. Yang, Symbol recognition via statistical integration of pixel-level constraint histograms: a new descriptor. IEEE Trans. Pattern Anal. Mach. Intell. **27**(2), 278–281 (2005)
75. W. Zhang, L. Wenyin, K. Zhang, Symbol recognition with kernel density matching. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2020–2024 (2006)
76. D. Thanh Ha, *Sparse Representation over Learned Dictionary for Document Analysis*. Ph.D. thesis, LORIA, Université de Lorraine, France (2014)
77. K.C. Santosh, *Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis)*. Ph.D. thesis, University of Lorraine, France (2011)
78. M. Rusiñol, J. Lladós, *Symbol Spotting in Digital Libraries: Focused Retrieval over Graphic-rich Document Collections* (Springer, London, 2010)
79. M. Delalandre, E. Valveny, T. Pridmore, D. Karatzas, Generation of synthetic documents for performance evaluation of symbol recognition and spotting systems. Int. J. Doc. Anal. Recogn. **13**(3), 187–207 (2010)
80. O. Ramos Terrades, S. Tabbone, E. Valveny, A review of shape descriptors for document analysis, in *Proceedings of International Conference on Document Analysis and Recognition* (2007), pp. 227–231
81. P. Dosch, E. Valveny, Report on the second symbol recognition contest, in *Graphics Recognition. Ten Years Review and Future Perspectives, 6th Internation Workshop, GREC 2005, Revised Selected Papers* (2005), pp. 381–397
82. L. Wendling, J. Rendek, P. Matsakis, Selection of suitable set of decision rules using choquet integral (2008), pp. 947–955
83. J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *Graphics Recognition – Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon. *Lecture Notes in Computer Science*, vol. 2390 (Springer, 2002), pp. 104–127
84. P. Chi Yuen, G.-C. Feng, Y. Yan Tang, Printed chinese character similarity measurement using ring projection and distance transform. Int. J. Pattern Recognit Artif Intell. **12**(2), 209–221 (1998)
85. J.P. Salmon, L. Wendling, S. Tabbone, Improving the recognition by integrating the combination of descriptors. Int. J. Doc. Anal. Recogn. **9**(1), 3–12 (2007)
86. O. Ramos Terrades, E. Valveny, S. Tabbone, On the combination of ridgelets descriptors for symbol recognition, in *Graphics Recognition. Recent Advances and New Opportunities*,ed. by L. Wenyin, J. Lladós, J.-M. Ogier. *Lecture Notes in Computer Science*, vol. 5046 (Springer, 2008), pp. 40–50
87. S. Barrat, S. Tabbone, A bayesian network for combining descriptors: application to symbol recognition. Int. J. Doc. Anal. Recogn. **13**(1), 65–75 (2010)
88. O. Ramos Terrades, E. Valveny, S. Tabbone, Optimal classifier fusion in a non-bayesian probabilistic framework. IEEE Trans. Pattern Anal. Mach. Intell. **31**(9), 1630–1644 (2009)

89. N. Nayef, T.M. Breuel, Statistical grouping for segmenting symbols parts from line drawings, with application to symbol spotting, in *Proceedings of International Conference on Document Analysis and Recognition* (2011), pp. 364–368
90. N. Nayef, T.M. Breuel, Efficient symbol retrieval by building a symbol index from a collection of line drawings, in *Document Recognition and Retrieval*, ed. by R. Zanibbi, B. Coüasnon. SPIE Proceedings, vol. 8658 (SPIE, 2013)
91. E. Valveny, M. Delalandre, R. Raveaux, B. Lamiroy, Report on the symbol recognition and spotting contest, in *Graphics Recognition. New Trends and Challenges - 9th International Workshop, Revised Selected Papers* (2011), pp. 198–207
92. N. Nacereddine, S. Tabbone, D. Ziou, L. Hamami, Shape-based image retrieval using a new descriptor based on the radon and wavelet transforms, in *Proceedings of the IAPR International Conference on Pattern Recognition* (2010), pp. 1997–2000
93. R. Souvenir, K. Parrigan, Viewpoint manifolds for action recognition. EURASIP J. Image Video Process. **1–13**, 2009 (2009)
94. Z. Ali Khan, W. Sohn, A hierarchical abnormal human activity recognition system based on r-transform and kernel discriminant analysis for elderly health care. Computing **95**(2), 109–127 (2013)
95. T.V. Hoang, S. Tabbone, The generalization of the r-transform for invariant pattern representation. Pattern Recogn. **45**(6), 2145–2163 (2012)
96. T.V. Hoang, S. Tabbone, Invariant pattern recognition using the rfm descriptor. Pattern Recogn. **45**(1), 271–284 (2012)
97. C. Charrier, L.T. Maloney, H. Cherifi, K. Knoblauch, S. Lô, Maximum likelihood difference scaling of image quality in compression-degraded images. J. Opt. Soc. Am. A **24**(11), 3418–3426 (2007)
98. M. Hasegawa, S. Tabbone, A shape descriptor combining logarithmic-scale histogram of radon transform and phase-only correlation function, in *Proceedings of International Conference on Document Analysis and Recognition* (2011), pp. 182–186
99. J. Coetzer, *Off-line Signature Verification*. Ph.d. thesis, Department of Applied Mathematics, University of Stellenbosch (2005)
100. K.C. Santosh, Character recognition based on dtw-radon, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2011), pp. 264–268
101. P. Toft, *The Radon Transform - Theory and Implementation*. Ph.D. Thesis, Department of Mathematical Modelling, Technical University of Denmark (1996)
102. J.-K. Kourosh, S.-Z. Hamid, Radon transform orientation estimation for rotation invariant texture analysis. IEEE Trans. Pattern Anal. Mach. Intell. **27**(6), 1004–1008 (2005)
103. GREC, International symbol recognition contest at grec2003 (2003)
104. J.B. Kruskall, M. Liberman, The symmetric time warping algorithm: from continuous to discrete, in *Time Warps, String Edits and Macromolecules: The Theory and Practice of String Comparison* (Addison-Wesley, 1983), pp. 125–161
105. M. Tooley, D. Wyatt, *Aircraft Electrical and Electronic Systems: Principles* (Operation and Maintenance, Aircraft engineering principles and practice (Butterworth-Heinemann, 2008)
106. E. Valveny, Ph. Dosch, Symbol recognition contest: a synthesis, in *Graphics Recognition: Recent Advances and Perspectives – Selected papers from GREC'03*, ed. by J. Lladós, Y.B. Kwon *Lecture Notes in Computer Science*, vol. 3088 (Springer, 2004), pp. 368–385
107. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recogn. Lett. **33**(3), 331–341 (2012)
108. K.C. Santosh, L. Wendling, B. Lamiroy, Relation bag-of-features for symbol retrieval, in *12th International Conference on Document Analysis and Recognition* (2013), pp. 768–772
109. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. Artif. Intell. **28**(06), 1450017 (2014)
110. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
111. M.S. Kankanhalli, B.M. Mehtre, W. Jian, Kang, Cluster-based color matching for image retrieval. Pattern Recogn. **29**, 701–708 (1995)

Chapter 5
Structural Approaches

5.1 Context

For graphics recognition, we have observed that statistical pattern recognition approaches are found to be suitable for isolated graphical symbols, regardless of noise, deformation, degradation, and occlusion. Further, pattern can be handwritten and/or machine-printed. However, they may not work for complex and composite graphical symbols. In such a case, another idea is to decompose the symbols into either vector-based primitives, such as points, segments, lines, and arcs, or into meaningful parts (e.g., circles, triangles, and rectangles). These are considered as the peculiar structure of technical documents. These structures (low-level and high-level) can be considered as a set of primitives, which can naturally induce the use of both structural and syntactic approaches [1–4] to recognize graphical symbols and graphical elements of particular interests. Note that graphical elements can be just a portion of the complete graphical symbol. Recognition a part of the whole graphical symbol can help understand the graphical symbol recognition process. In high-level definition, we call it graphical symbol/element spotting.

The choice depends on how robust the representation can be made as well as on the skills to make algorithm faster (considering real-world projects). Since both approaches use primitives (as basic building blocks), in what follows, we first discuss primitive extraction. We can then discuss how can they be related/connected to each other. Connecting them could possibly be handled by spatial relations. Visual primitives and their possible relations (pairwise) introduce structural and syntactic approaches for graphical symbol recognition, retrieval, and spotting purposes. For specific applications, previously reported works can be referred [5–8].

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_5

5.2 Visual Primitives

If we consider a raster data, where a graphical symbol is naturally represented by the set of pixels. In this context, let us start with a few examples, where one can have usefulness of the local descriptors, and key point selection is the major concern. Key point selection at the corner can help understand/detect corners in addition to their orientations, such as corners facing northeast and southeast, for instance. Vectorization could be another interesting issue, such as dashed-line detection or just a straight line. Curve (in engineering drawing) can be another important features to be exploited. Since an arc can complement/support curve detection, arc detection could help recognize the structure of the graphical symbol. Further, high-level primitives, such as circle and thick component, can proceed with graphical symbol/element recognition much faster than that are coming from a collection of low-level primitives. Let us be specific, thick component refers to the filled region that represents diode in electrical circuit diagram.

In brief, let us provide a few issues with examples. For raster data, key point selection with the use of the local descriptors can be an issue to be discussed. It is, however, not trivial that which local descriptors are more appropriate and how their performances depend on the selection of key points or regions, since the use of descriptors varies from one dataset to another. This means that the use of local descriptors can be application dependent. Comprehensive discussions can be found in [9] for any image recognition/understanding problem.

(a) Vectorization [10] is the task of extracting primitives like simple lines [11, 12] and arcs [13–15] including geometric primitives, such as loops and contours or simple shapes like circles and rectangles.
In technical documents, these primitives are considered as primary elements as they can represent a complete graphical symbol structure [16–18]. These primitives serve a common basis or basic building blocks for both structural and syntactic approaches. It is important to notice that such a set of primitive (meaningful) extraction is not trivial since problems (studied image samples) vary widely.
For line extraction, the Hough transform [19, 20] can be used. With this, applications are reasonably limited. For example, in case of degraded scanned document, it suffers from high computational cost. Several approaches are based on the extraction of strings from the skeleton [21, 22] having well-known distortion problems but they require a correct localization of junctions (points) [23–26] by matching opposite edges [27]. These solutions are limited since the concept is sensitive to the complexity of processed shapes. Motivated by this, the orthogonal zig-zag algorithm [28] and adaptations [20, 29] were proposed to make line extraction efficient.

(b) Curves and arcs are taken as another subject of research interest. Digital curves have been extracted based on the polygonal approximation in the form of skeleton strings (using the Haar transform, for instance) [30–33]. The method leads naturally to a loss of information (or accuracy) with respect to the initial structure, but it offers implementations simplicity for the subsequent algorithms.

Other approaches that follow the initial curve are more accurate [34–36]. However, some of them follow dynamic programing concept and they are slow (processing time). In a few cases, code optimizations can help [37–39] speed up the process.

Few methods that are focussed on key points help the segmentation process since key points can be considered as seeds [40–43]. These approaches often require threshold initialization, due to which their performance varies as the type of the application changes. This means that developing generic algorithms is not trivial. It is also difficult to combine several different approaches to detect both arcs and segments that are fundamental to ensure accurate symbol recognition. Therefore, avoiding over-segmentation is required (see Fig. 5.1, for example). If not, it may generate several unwanted small components for further processing, i.e., matching in recognition step [44].

At this point, let us discuss on potential shortcomings. In arc-fitting methods [14, 26, 40, 45], the main drawbacks can be from noise and distortions that lead to local errors, i.e., over-segmentation is always possible. Other approaches that are built from the Hough transform [46, 47] often produce accurate results in the presence of distortion. But, like we have discussed earlier, manual threshold initialization is the must. The reason behind this is because they are sensitive to noise. Moreover, they suffer from expensive computational cost. Methods that are based on stepwise arc extension can improve segmentation by studying specific arcs [13, 48, 49], and remain more stable in more cases.

At this point, we observed that vector data require the use of robust extraction operators to ensure analysis and correct understanding of documents [50–52]. Most of these approaches have been compared in GREC contests: 2001 and 2003 [45, 53–55], for example. The generic method RANVEC [26, 54] overcomes other methods in most of the problems. One of the drawbacks of this method is that it may omit small parts since it selects points randomly, aiming to reduce the computational time. For more detailed information about GREC contests, refer to Chap. 2.

(c) Other primitives, such as thick components or filled regions, circles, corners, and extremities (i.e., loose end points) can be extracted using classical image analysis operators as reported in previous works [15, 57–59]. Figure 5.2 shows a few examples. As mentioned in the PhD thesis report [5], we can now describe a few of them as follows, in addition to recent developments:

- For thick components, it uses standard skeletonization using chamfer distance and computes the histogram of line thicknesses; and an optimal cut value is computed from the histogram to distinguish between thick zones and thin zones.
- For circle and arc, as in [15], the random sample consensus minimization (RANSAC) scheme is used to detect circles and arcs. The performance of the method was demonstrated (as the winner) in the arc segmentation contest on GREC in 2011 [60].

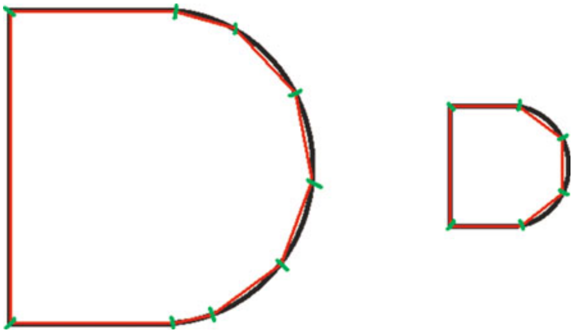


Fig. 5.1 An example showing over-segmentation [7]. It illustrates two identical shapes (symbols) in different scales produces different number of segments

Using gradient-direction-based segmentation and direct least-square fitting, a fast and accurate circle detection algorithm, reported in [61], outperforms the circular Hough transform [62], the randomized circle Hough transform [63], and the fast circle detection [64] in both processing speed and detection precision. Moreover, very recently, as reported in [65], their method also outperforms RANSAC [15] in arc detection precision and speed. In [66], very recent arc detection contest was reported.

- Considering technical documents (rich in graphics), a common drawback is error-prone raster-to-vector conversion. For example, primitive types such as arc and corner are not extracted (as expected) as the degradation or noise level increases. However, robust vectorization exists in line drawing images (engineering drawings, for instance) [25, 26, 67]. The use of such low-level primitives varies widely in accordance with the complexity of the symbol.

5.3 Spatial Relations

Spatial reasoning is regarded as a central skill to many human tasks, as being able to communicate about the space. A common and natural way to share spatial information is through the use of spatial predicates [68] such as *Left of* and *Right of*, in order to derive relationship between the spatial entities.

To handle image recognition, partial recognition of visual primitives used to guide the recognition of remaining parts within it [69]. It is based on the question, i.e., effect of spatial relations on recognition performance [70, 71]. A quick explanation of this can be illustrated from Figs. 5.3, 5.4 and 5.5. Can we connect (via relations) all objects to see whether it is possible to describe image? In Fig. 5.3, without having *a priori* knowledge about objects (their labels), one cannot make a difference between two images. We can, however, take advantage of intra- and inter-object relations to describe images. A prototype is shown in Fig. 5.4. Such a description can make two images different. This could be computationally expensive since we have compute

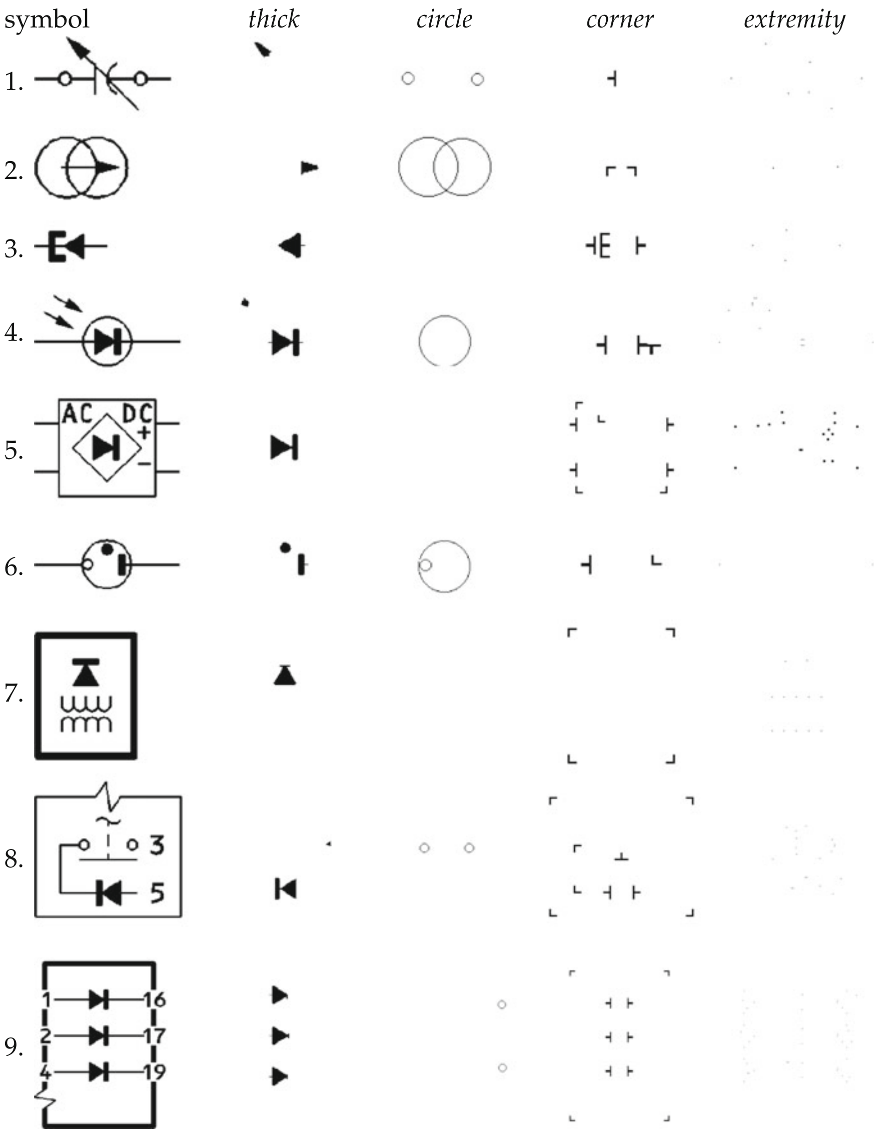


Fig. 5.2 Few examples showing visual primitive extraction. Visual primitives are thick (or filled) components, circles, and corners [5]. The samples are taken from FRESH dataset [56]

two different layers of relation: intra- and inter-object. Intra-object relations can describe the shape of the object and inter-object can help understand the arrangements in the complete image. Another idea could be based on region segmentation (see Fig. 5.5). In this prototype, we observe that an object can be decomposed into a set of regions, i.e., regions-of-interest that are then used for description via relations. In

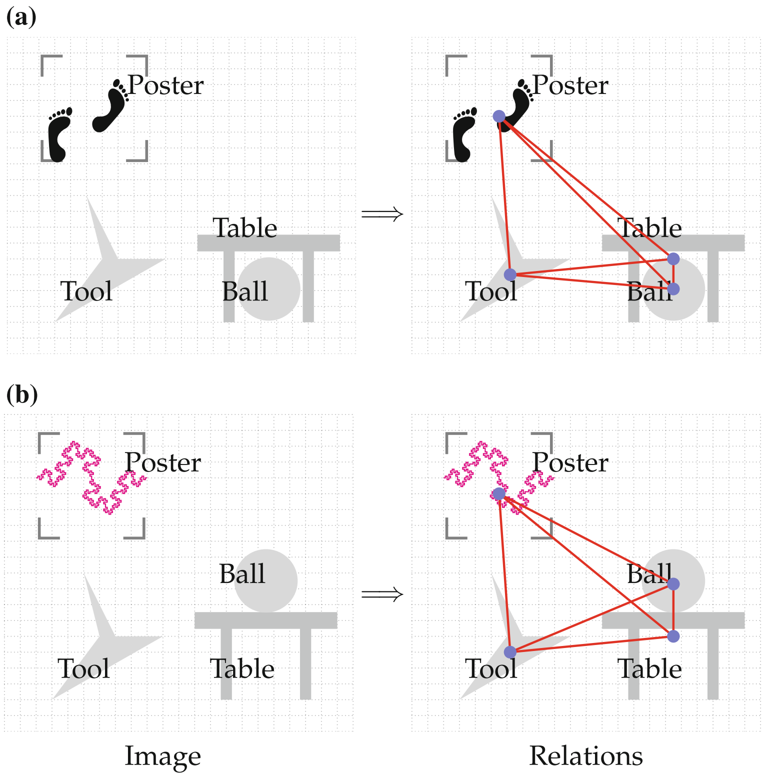


Fig. 5.3 Can we just rely on relations for an image description? It can help provide spatial arrangements (of the objects) but not their features/properties. Note that posters are different from one image to another. Centroids (of the objects) are used to compute spatial relations

the domain of graphics recognition, the concept has been widely used. Section 5.4 will provide more detailed explanation (see Figs. 5.11 and 5.12). Before that, let us first discuss types of relations, and their merits and demerits.

Not a surprising, in document image analysis, relations are used for analysis of architectural documents and for automatically recognition of models [72] and the graphical drawings understanding of scanned color map documents [73] or to define efficient retrieval methods [74–77].

In the following, we first briefly outline spatial relations, their types, properties, as well as their appropriate applications. After that, detailed study about the impact of spatial relations for document image analysis (graphics recognition, in particular) will be explained.

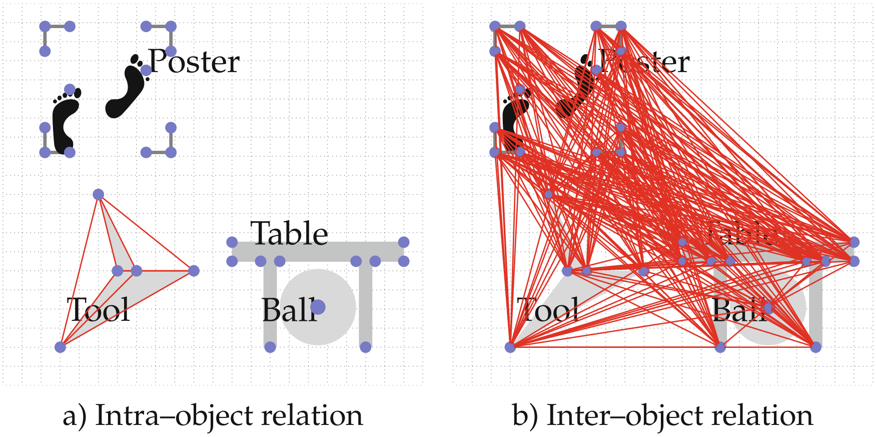


Fig. 5.4 Image can be described via the use of spatial relations: intra- and inter-object. Intra-object graph can help describe object on the whole, and that makes two posters are different as shown in Fig. 5.3. Inter-object relations are shown as before

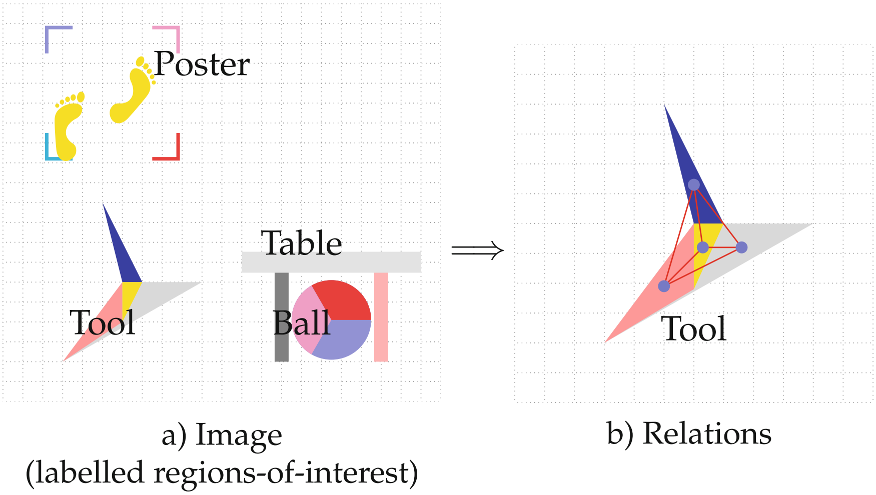


Fig. 5.5 Image can be described via the use of spatial relations: intra- and inter-object, where regions-of-interest (ROIs) are used. Each color represents a region to be used for computing spatial relations

5.3.1 Types of Spatial Relations

In [78], authors provide one of the first consistent studies of spatial relations and their variations according to the context. An important family of spatial relations and associated properties came from Freeman [79], and are grouped as follows:

(a) topological relations,

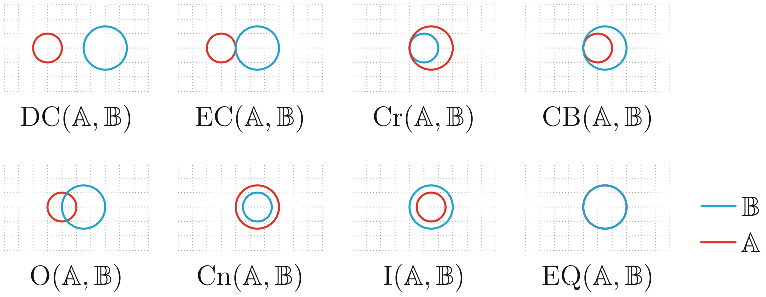


Fig. 5.6 Possible topological relations between two objects $\mathbb{A}$ and $\mathbb{B}$

(b) metric relations, and
(c) directional relations.

Topological relations describe neighborhoods and incidence, such as *disconnected* and *externally connected*, metric describes distance relations like *near* and *far*, and directional relations provide order in space, such as *north*, *south*, and *east*.

(a) *Topological relations*:
In connection with [80], basic topological relations close to human understanding are *disconnected* (DC), *externally connected* (EC), *covers* (Cr) or *covered by* (CB), *contains* (Cn) or *inside* (I), and *equal* (EQ). Figure 5.6 shows an illustration of it. In this illustration, we can observe that topological relations satisfy affine transformation invariant properties.

(b) *Metric relations*:
It provides an idea of distance between two spatial objects. Consider a metric on a set X is a function (called the distance function or simply distance), then

$$d : X \times X \rightarrow \mathbb{R},$$

where $\mathbb{R}$ is the set of real numbers. For all x, y, and z in X, this function is required to satisfy the following conditions (c1–c4):

(c1) $d(x, y) \geq 0$, i.e., non-negativity,
(c2) $d(x, y) = 0$ if and only if $x = y$,
(c3) $d(x, y) = d(y, x)$, i.e., symmetry, and
(c4) $d(x, z) \leq d(x, y) + d(y, z)$, i.e., triangle inequality.

Based on it, many modifications have been made according to the applications.

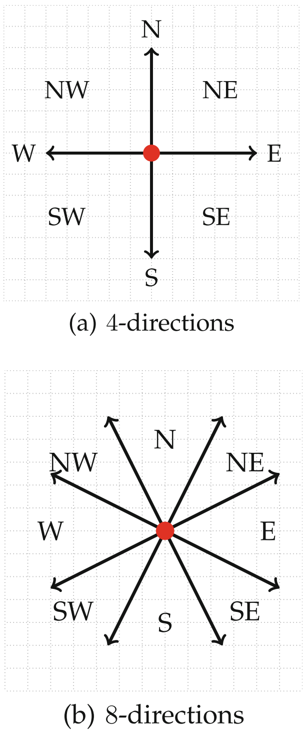
Computer representation of geospatial information has been motivated by proximity relations such as *nearness* and *locality*, as described in [81]. For example, *nearness* is derived from relative distance, i.e., *relative distance*$(x, y) = \frac{d(x,y)}{\mu_c}$ where μ_c is the mean distance measured from the center.

(c) *Directional relations*:
In general, they provide an idea of *orientation* of the primary spatial objects with

respect to a reference. Each object is represented by one or more representative points, and the space is partitioned using these points. The relation is then determined using the partitions to find where the object representative points are. Depending on the concept of partitioning, there are several different ways to handle directional relations between the spatial objects. In the following, we provide some of the fundamental concepts or models.

(i) *Angle-based model*
Angle-based relational models provide a true orientation of spatial objects. Two fundamental models fall under this category.

- *Cone-shaped model*: The relations can be approximated by their centers based on discretized angle [82, 83], i.e., $\angle(C_{\mathbb{A}}, C_{\mathbb{B}})$ between the objects $\mathbb{A}$ and $\mathbb{B}$. It is sometimes also called the bi-center model. It provides several different configurations based on star calculus [84]. In Fig. 5.7, one can have an idea of progressive refinement of bi-center model from 4 to 8 directions and so on. This model possesses the following shortcomings.
  – Relations remain unchanged unless there exists significant separation.



(a) 4-directions

(b) 8-directions

Fig. 5.7 Star calculus via bi-center model via angle-based theory. It shows a cone-based method to compute relations between the studied objects (primitives). The star calculus can be extended with a small angular step

  - It does not take shape and size into account. Since it is robust to small variations of shape and size, one cannot guarantee that the centroid falls within the spatial object.
  - It does not carry topological information: objects having *inside* or *contain* topological configurations, for instance, yield ambiguous spatial predicates.
  - It provides no directional relation in case of centroid coincidence from two studied objects (even for two different shapes).

  It is best suited when the studied objects are far from each other.

- *Angle histogram*: Such an approach considers all pixels where the cone-shaped model only took the centroid into account [85]. As a consequence, their computational cost increases dramatically.

  Let us consider two objects $\mathbb{A}$ and $\mathbb{B}$ as the sets of their pixels:

$$\mathbb{A} = \{a_i\}_{i=1\ldots m} \text{ and } \mathbb{B} = \left\{b_j\right\}_{j=1\ldots n}.$$

  The $m \times n$ pairs of points allow for the computation of a set of angles $\theta_{i,j}$ between each (a_i, b_j). The histogram $\mathscr{H}$ representing the frequency of occurrence of each angle f_θ can then be formulated as

$$\mathscr{H}_\theta(\mathbb{A}, \mathbb{B}) = [\theta, f_\theta].$$

  For a simplicity, histogram values can be aggregated into a single value.

  The major difference with bi-center model lies in the fact that the averaging is made on the objects' points for the centroid method while it is applied after angle computation in the aggregation method. If the objects are far from each other, this averaging converges to bi-centercenter model.

(ii) *Projection model*:

The projection model uses the classical minimum boundary rectangle (MBR) model [86]. Figure 5.8 shows the MBR model and its iconic vertical and horizontal projections, regardless of the compactness of the objects. Compacity is defined as the percentage of the spatial object in the MBR. Such a partitioning of the space is dynamic according to the shape and size variation of the reference object [87].

The following properties can be summarized.

- MBR is only appropriate as long as spatial objects are regular. This means that it depends on compactness. Compacity of more than 0.80 is found to be regular.
- False overlapping is possible. It misleads results in case of no actual intersection of the spatial objects (see Fig. 5.9).

Having fundamental concepts of all three basic families of spatial relations, can we provide knowledge representations in one particular problem?

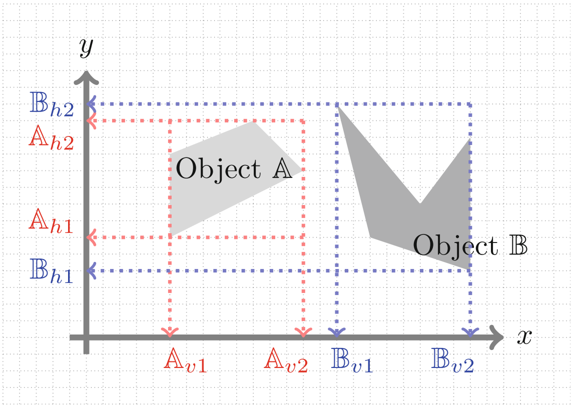


Fig. 5.8 MBR: a complete illustration showing horizontal and vertical projections ($\mathbb{X}_{h1}$ and $\mathbb{X}_{h2}$ refer to horizontal projections, and $\mathbb{X}_{v1}$ and $\mathbb{X}_{v2}$ refer to vertical projections from both $\mathbb{A}$ and $\mathbb{B}$)

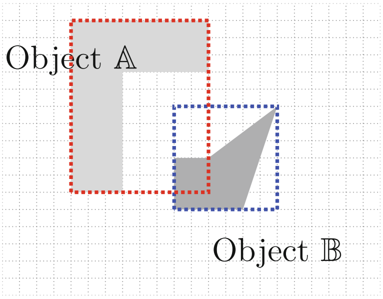


Fig. 5.9 False overlapping, example (where compacity($\mathbb{A}$) $= 0.56$)

5.3.2 Can We Quantify Spatial Relations?

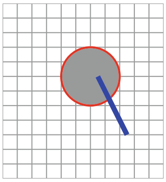
Now the question is can be precise enough to deal quantitative spatial reasoning for image recognition, which is beyond geographical structures (or GIS)? Another way to categorize spatial reasoning [88, 89] is either based on

(a) qualitative or
(b) quantitative knowledge representation.

The former one conveys Boolean spatial information, i.e., 1 for the presence of spatial object and 0 otherwise. The latter one is often based on fuzzy set theory [90] allowing a better managing of the ambiguous aspect of spatial relations. However, their impacts vary equally according to the nature of applications. Let us have two examples so that both ideas can be explained, and help us find the differences between them.

(a) In Fig. 5.8, consider an object $\mathbb{A}$ as the reference, and we have the following relations:

Fig. 5.10 Inner transversal splitting (INS) example. The concept is taken from the previous work [91]



$$\text{INS} = \frac{length(\text{Line} \cap \text{Region})}{length(\text{Line})}$$

$$\text{Qualitative relation}(\mathbb{B}, \mathbb{A}) = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \text{ and}$$

$$\text{Quantitative relation}(\mathbb{B}, \mathbb{A}) = \begin{bmatrix} 0 & 0 & 0.005 \\ 0 & 0 & 0.880 \\ 0 & 0 & 0.115 \end{bmatrix}.$$

This means that object $\mathbb{B}$ is found to be extended from *right bottom* to *right top* with respect to object $\mathbb{A}$.

(b) In another example, one can consider the complexity in drawing images such as ladder-like sketches. For example, *overlap*$(line_{\mathbb{A}}, rectangle_{\mathbb{B}})$ does not answer about “how much”. Therefore, metrical details are necessary as explained in [91], for instance. Figure 5.10 shows an example of it.

Overall, metrical details provide precision and of course, more confidence in recognition. However, the selection of either qualitative or quantitative spatial representation can be summarized as follows:

- Qualitative interpretation provides spatial relations more close to natural language as used in spatial predicates like *right*, *left*. Qualitative knowledge is usually cheaper since it does not need to compute percentage value.
- On the other hand, quantitative spatial reasoning is chosen in cases where it needs natural instead of all-or-none relations [79]. Consequently, fuzzy concepts have been introduced in several different applications since they are directly related to shape and size information and are comparable to human perception. Angle histograms rather than just a single angle value is one of the basic examples. Similarly, fuzzy landscape based on fuzzy morphological operators [92] and force-histogram approaches have been popularly used such as force histograms [93]. If there is uncertainty, then it is inherently suited for fuzziness [94].

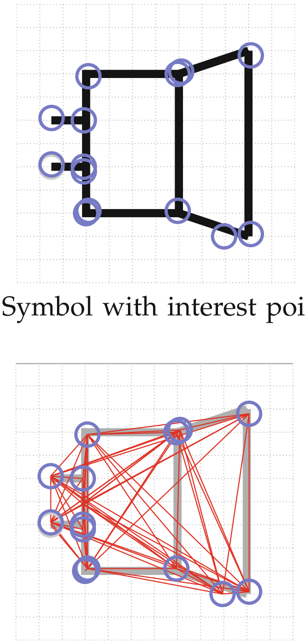
5.4 Structural Approaches for Graphics Recognition

In brief, structural approaches are based on symbolic data structures such as strings, trees, and graphs. Graphs are most commonly used while strings and trees are always included as special cases. In document analysis, the most recent advancement in graph-based pattern recognition is presented in [95]. The formal concept of graphs can be found in [96].

How can we make structural approaches different from statistical ones? Unlike statistical approaches, structural approaches provide a powerful representation, conveying how parts are connected to each other, while also preserving generality and extensibility [5, 97–100]. Graph-based or graph-like representation provides an abstract concept of the studied image. Let us elaborate this with some examples. In [101], authors introduced the notion of interest points by considering corners and junctions. Later, it has been represented using local descriptors such as SIFT features [102]. In [103], for example, local descriptor (Harris–Laplace detector [104]) is used to build a proximity graph for any studied symbol. Figure 5.11 shows an example of it. In [105], adjacency relations between the segmented regions have been described. Skeletal graph for shape representation is another example. It uses skeleton points, which are categorized into three families: junction, end, and branch points. Considering graphical symbol recognition, in [106], a skeletal graph is used to represent the symbol from electrical diagrams. For graph matching, bounded search is used to select the pose of the graph such as rotation, translation, and scale for a minimum error transformation. It is entirely based on probabilistic models. In [107], graphs are used for building a model-based scheme for recognizing hand-drawn symbols in schematic diagrams. To construct the graph, as said before, endpoints, junctions, and crossings are represented by vertices attributed with the number of neighbors and the angles between incident edges. The edges represent connecting lines in the drawing attributed with the length and curvature of the respective line. Such graph-based representation schemes are not limited; they vary widely, such as attributed relational graph (ARG) [108–110], region adjacency graphs (RAG) [111, 112], constraint networks [113], deformable templates [114], and proximity graph [103]. Similarly, other forms of graph-like ARG (specifically designed for symbol representation) [115] provide fundamental parameters related to structural approaches. Figure 5.12 shows an example of a line graph. On the whole, they vary from one application to another, i.e., one representation does not fit for all [116, 117].

Structural approaches are particularly well suited for recognizing complex and composite graphical symbols [5, 118]. Recognizing region-of-interest (graphical symbol, for instance) in a technical document refers to the identification/detection of a part of the graph. The process is known by the name subgraph isomorphism, which is crucial in any real-world context [119, 120]. This, on the other hand, as said before, leads to a very high computational cost (NP-hard problem: polynomial time) [96, 121], which is often the case when complex and composite symbols are taken for evaluation. Further, variability of the size of graphs that can be due to presence of noise and possible distortions leads to computational complexity in matching. Besides, their common drawback comes from error-prone raster-to-vector conversion. This makes symbol representation weak but, varies with the application.

In the framework of stroke-based hand-drawn symbol recognition, several studies have been presented in [122, 123]. The first study is related to template-based matching and another one uses ARGs where the vertices represent geometric primitives



(a) Symbol with interest points

(b) Proximity graph

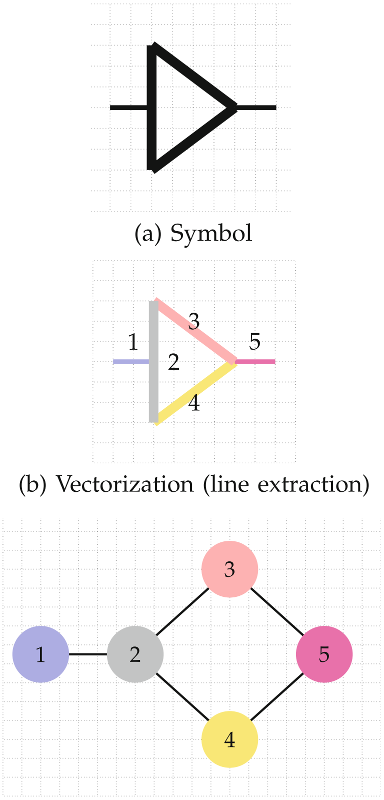
Fig. 5.11 Proximity graph representation using interest points (circle in blue) using local descriptor [5], where edges are represented in red (see Fig. 5.4 for a generic concept)

like lines and arcs (based on their shapes) and the edges represent the geometric relationships between them. For matching, it is primarily based on graph matching or graph isomorphism detection presented in [124]. The work is conceptually similar to [125], which was extended from previous works reported in [126, 127]. These approaches perform well as long as vertices are well separated since they are taken from online strokes. On the whole, this presents an idea of how vectorization difficulties are avoided.

In [128, 129], their interest is to map an inaccurate isomorphic structure to address noise artifacts and distorted data by incorporating cost functions for deletion, insertion, and node/edge modification. We know that the methods are still sensitive to noise and are suffering from heavy time computational cost even after the integration of the statistical assumptions using error-tolerant features when searching subgraphs [112, 130], for instance. For the same problem, various heuristics are still employed, but they do not guarantee to show significant difference. They, however, aim to have a closer to a so-called optimal solution as long as the problem is defined with some constraints.

Considering the time complexity issue, several works are more focused on computing symbol signatures by taking regions-of-interest (ROIs) in the document image [131–133]. These methods aim to provide faster matching in comparison to general graph matching. The performance of such methods is based on how accurate the ROIs are extracted. This means that the methods fail when ROIs do not carry

Fig. 5.12 An example illustrating **a** a symbol, **b** line extraction, and **c** its corresponding relational graph [5] (see Fig. 5.5 for a generic concept)



(a) Symbol

(b) Vectorization (line extraction)

(c) Relational graph

symbols. Further, in [134], author aimed to address how efficiently computational cost can be reduced through the use of graph-based structural pattern recognition approaches. The work has been developed to represent an image by a graph using the state-of-the-art methods but, a new technique named fuzzy multilevel graph embedding (FMGE) is used to transform that graph to numeric feature vectors [135]. As a consequence, it empowers the structural pattern recognition approaches by utilizing statistical pattern recognition tools. Such a graph is used to perform symbol (line drawing) recognition and spotting. While transforming the graphs to numeric feature vectors, there may a loss of information. As a consequence, the FMGE method can be compared with inexact methods, but it may be less accurate than exact methods.

In [136], authors proposed a symbol spotting technique using graph serialization to reduce the usual computational complexity of graph matching. As said before, graphs are used to represent the documents and a (sub)graph matching technique is used to detect the symbols in them. Serialization of graphs is performed by computing acyclic graph paths between each pair of connected nodes. Graph paths are one-dimensional structures of graphs which are less expensive in terms of computation.

At the same time, they enable robust localization even in the presence of noise and distortion. For large graph databases, they propose a graph factorization approach to index, which is intended to create a unified indexed structure. Once graph paths are extracted, the entire database of graphical documents is indexed in hash tables by locality sensitive hashing (LSH) of shape descriptors of the paths. The hashing data structure aims to execute an approximate k-NN search in a sub-linear time.

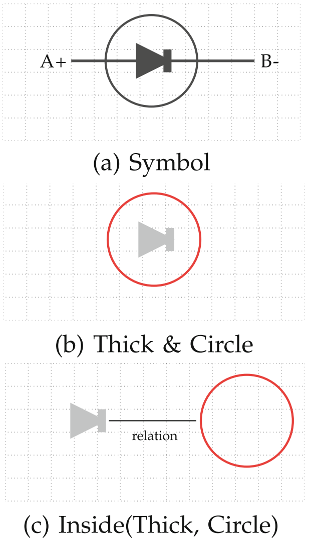
Other methods are based on the relaxation principle based on constraint propagation for matching nodes. Besides high computational cost, another main drawback is that it does not guarantee robustness for correct local matching. These discrete approaches, where a label is associated with each primitive, allow a local focusing on inconsistent matching. Such a principle has been used to carry electrical symbol recognition [137, 138]. Later, Wilson and Handcock have extended the discrete relaxation by introducing a Bayesian model [139]. The probabilistic relaxation assigns each node a probability measure according to the constraints. They are then iteratively updated to maximize a measure of consistency on the whole [140–142]. Fuzzy concept has also been in use on the relaxation mechanism for handling uncertain data [143, 144]. Graph-based method allows invariants and is independent of labels [145].

Very recently, in [146, 147], authors introduced an interesting approach where a Galois lattice is used to classify structural signatures that are extracted using the Hough transform. These structural signatures are based on a topological graph, where there are only five topological relations computed between the segments (based on connected and disconnected topological configurations). As reported in the paper, the Galois lattice-based classification is robust to noise. However, the performance is inconsistent when symbols are found to be connected with other graphical elements or texts in the whole document image. Later, the advancement has been reported in [148, 149].

In [150], authors addressed the problem of symbol spotting in scanned and vectorized line drawings. A set of primitives index the structure of the symbol it is composed of. Such an indexing is used to retrieve similar primitives from the database. Primitives are encoded in terms of attributed strings representing closed regions. Similar strings are clustered in a lookup table so that the set of median strings act as indexing keys. A voting scheme formulates hypothesis in certain locations of the line drawing image where there is a high presence of regions similar to the queried ones, and therefore, a high probability to find the queried graphical symbol. As reported in the paper, the method is found to be robust to noise and distortion, which are introduced by the scanning and raster-to-vector processes. A comprehensive study has been reported in [151].

5.5 Spatial Relations on Graphics Recognition

Effect of spatial relations on recognition performance was examined comprehensively for image and/or scene understanding [69], document analysis, and recognition



(a) Symbol

(b) Thick & Circle

(c) Inside(Thick, Circle)

Fig. 5.13 Example: use of spatial relation in the context of symbol recognition. Use spatial predicates, such as *inside* make easy to understand the relation between two primitives: *thick* (or filled component) and *circle* that compose a symbol

tasks [152, 153]. As mentioned earlier, spatial relations can be either topological [80, 154], directional [83, 93], and metric in nature. Their choice/selection depends on type of application, i.e., studied sample: how complex it is. For example, in [125], topological configurations are handled with a few predicates like (a) *intersection*, (b) *interconnection*, (c) *tangency*, (d) *parallelism*, and (e) *concentricity* expressed with standard topological relations as described in [154]. However, more often, we have *disjoint*, *touch*, *overlap*, *contain/inside*, *cover/covered by*, and *equal* (see Fig. 5.6).

In a similar way, various directional relation models have been developed for a wide range of different situations. A summary of what has been discussed in the earlier section can be itemized as follows:

- If the objects are far enough from each other, their relations can be approximated by their center based on the discretized angle: bi-center model [82, 83].
- If they are neither too far nor too close, relations can be approximated by their minimum bounding rectangle (MBR) as long as they are regular [86, 155, 156]. But the quality of the MBR depends on compactness (i.e., Compactness = $\frac{Area(\mathbb{A})}{Area(MBR(\mathbb{A}))}$) of the MBR tile.
- Approaches like angle histograms [85] tend to be more capable of dealing with overlapping, something the previous approaches have difficulties with. However, since they consider all pixels, their computational cost increases dramatically. In [157, 158], authors have shown an example of a piece of engineering work about how the MBR and the angle-based theory recognize graphical symbols. Other methods, like for instance, force histograms [93], use pairs of longitudinal sections instead of pairs of points, also at the cost of high time complexity. Use of fuzzy landscapes [92] that is based on fuzzy morphological operators can be another idea.

At this point, the question is, "can we think of integrating different two types of spatial relations: topology and direction?" Have we observed previous works address the issue?

Previously mentioned approaches address only either topological or directional relations. Managing both comes at high computational costs. Even then, no existing model fully integrates topology with directions. They rather have various degrees of sensitivity to or awareness of topological relations. For graphics recognition, while methods like [125] focus on topological information only, the approach we will discuss is to see whether we can unify both topological and directional information and make one descriptor [158, 159] without adding significant running time cost.

Placing spatial relations in the context of recognition and symbol description [159], one should note that spatial relations also have a language-based component (related to human understanding, e.g., to the *right* of) that can be formalized in a mathematical way (e.g., the 512 relations of the 9−intersection model [154]). Therefore, qualitative and quantitative relations are interesting representations. For example, consider an object $\mathbb{A}$ extending from *right* (98%) to *top* (2%) with respect to $\mathbb{B}$ is expressed as *right* – *top*($\mathbb{A}$, $\mathbb{B}$). This spatial predicate remains unchanged up to a reasonable change of the objects' shape and position. Taking this into account, let us discuss natural relations than the all-or-none nature of standard relations [79].

Once again, to handle image recognition, partial recognition of primitives is used to guide the recognition of remaining parts within it [69]. In this context, for a complete image recognition, the effect of spatial relations between the primitives on recognition performance has to be determined [70, 71]. In graph-based pattern representation, connectivity that exists between the nodes must have meaningful information. Spatial reasoning has been considered as a central skill since it is a common and natural way to share spatial information through the use of spatial predicates [68] that ultimately derive relationship between the primitives. To illustrate such a concept, we refer to Fig. 5.13, where it shows how connectivity between the primitives must be meaningful for recognition. To compute spatial relations, a common angle-based theory based on star calculus is shown in Fig 5.7.

5.6 Can We Take Complex and Composite Graphical Symbols into Account?

Not just limited to star calculus model, in document image analysis, precise relations are required to analyze/understand scanned architectural and color map documents [160], recognize models [72], and to define efficient retrieval methods [75–77, 161]. Similarly, authors have shown the usefulness of relational indexing of vectorial primitives for line drawing images [162]. Considering the problem of symbol localization in real documents, composed of individual parts and constrained by spatial relations, as said before, global signal-based descriptors cannot be applied since they are, unfortunately, primarily designed for applications where symbols are isolated. Such a problem related to the segmentation/recognition paradigm [163], for instance, where an accurate segmentation of meaningful parts is expected. In this context (as shown in Fig. 4.3 of Chap. 4), we are required to formalize the possible links that exist between them to build a graph-like structure [159, 164, 165]. Considering such a real-world problem, these methods outperform the state-of-the-art methods used in graphical symbol recognition.

5.6.1 Symbol Recognition Using Spatial Relations

Following earlier section, let us take those graphical symbols as shown in Fig. 4.3 of Chap. 4. If we consider/follow the how structural approaches have been defined, we can easily come up with the idea that spatial relations can be computed between all possible pairs of the visual primitives as shown in Fig. 5.2 that compose a complete symbol. At the same time, we also need to think of time complexity. Note that the concept was primarily taken from previous works [5, 6, 159] in order just to provide a screenshot of how relations affect the recognition performance.

In what follows, we first have a clear concept of what spatial relations are and how can a graphical symbol be represented. Then, a series of experimental test results will be provided.

(a) *Visual primitives and spatial relations*:
A set of well-controlled visual primitives [166] are defined as shown in Fig. 5.2. To express the spatial relations, we compute a spatial signature R that exists between any two visual primitives: $\mathbb{A}$ and $\mathbb{B}$. To compute R, we are required have reference. For example, $\mathbb{A}$ is to the *right* of $\mathbb{B}$: $right(\mathbb{A}, \mathbb{B})$, where $\mathbb{B}$ is referenced. In this context, since the number of vocabulary types varies from one image to another, one cannot have a thumb rule for a reference selection. In this case, it is a wise idea to compute a unique reference point from each pair. After that, directional relations (qualitative and/or quantitative) with respect to the reference point can be computed. This can avoid potential ambiguity in reference selection.
A unique reference set $\mathbb{R}$ is defined by the topology of the MBRs of $\mathbb{A}$ and $\mathbb{B}$ and with the help of the 9-intersection model [154]. Using [80], $\mathbb{R}$ is either the common portion of two neighboring sides in the case of *disconnected* MBRs or the intersection in the case of *overlapping*, *equal* or otherwise *connected* MBRs. Based on the topology, $\mathbb{R}$ can be either a point, a line or a rectangle. Regardless the nature of the $\mathbb{R}$, its centroid point $\mathbb{R}_p$ is taken as a reference point so that spatial relation R between $\mathbb{A}$ and $\mathbb{B}$ can be computed.

For a given reference point $\mathbb{R}_p$, let us rotate a radial line with a regular interval of $\Theta = 2\pi/m$. As shown in Fig. 5.14, a radial line rotation generates a Boolean histogram H by intersecting object $\mathbb{X}$ ($\mathbb{A}$ or $\mathbb{B}$) in the space,

$$H(\mathbb{X}, \mathbb{R}_p) = [I(\mathbb{R}_p, j\Theta)]_{j=0,\ldots,m-1},$$

where

$$I(\mathbb{R}_p, \theta_i) = \begin{cases} 1 \text{ if } line(\mathbb{R}_p, \theta_i) \cap \mathbb{X} \neq \emptyset \\ 0 \text{ otherwise.} \end{cases}$$

Without loss of generality, such a histogram that can cover sectors defined by two successive angle values. Furthermore, rather than providing Boolean values (qualitative relations), one can take percentage of pixels into account of the whole object. Considering both objects: $\mathbb{A}$ and $\mathbb{B}$, spatial relational signature $R(\mathbb{X}, \mathbb{R}_p)$ is the set of both histograms:

$$R(\mathbb{X}, \mathbb{R}_p) = \{H(\mathbb{A}, \mathbb{R}_p), H(\mathbb{B}, \mathbb{R}_p)\}. \tag{5.1}$$

This means that relational signature/histogram can take object's shape and sizes into account, in addition to spatial features. Figure 5.15 provides a graphical illustration.

In brief, for each sector (made by two consecutive radial lines, see Fig. 5.14), histograms are computed for both visual primitives, i.e., counting the percentage of pixels of the studied primitives lying in it.

(b) Graph-based graphical symbol representation:

Since visual primitives types are fixed and labeled, one can compute spatial relations between the types (in terms of histograms described earlier). This tells us all that symbol can be represented as a complete ARG, where each vertex represents a distinct attribute type and the edges are labeled with a numerical expression of the spatial relations.

More formally, we express the ARG as a 4-tuple $G = (V, E, F_A, F_E)$ where

V is the set of vertices;

$E \subseteq V \times V$ is the set of graph edges;

$F_A : V \rightarrow A_V$ is a function assigning labeled attributes to the vertices where A_V is the set of attributes type set $\sum_{\mathbb{T}}$; and

$F_E : E \rightarrow \Re_E$ is a function assigning labels to the edges where R represents the spatial relation of the edge E . Note that R does not provide symmetry, $R(\mathbb{A}, \mathbb{B}) \neq R(\mathbb{B}, \mathbb{A})$. But, this can be solved by fixed ordering of V and $\mathbb{R}$ is not affected.

For any graphical symbol having three different types of attribute: $\{\mathbb{T}_1, \mathbb{T}_2, \mathbb{T}_3\}$, the following ARG representation can be made:

$$
\begin{aligned}
G = \{ & \\
V &= \{\mathbb{T}_1, \mathbb{T}_2, \mathbb{T}_3\}, \\
E &= \{(\mathbb{T}_1, \mathbb{T}_2), (\mathbb{T}_1, \mathbb{T}_3), (\mathbb{T}_2, \mathbb{T}_3)\}, \\
F_A &= \{(\mathbb{T}_1, \mathbb{T}_{circle}), (\mathbb{T}_2, \mathbb{T}_{corner}), (\mathbb{T}_3, \mathbb{T}_{extremity})\} \\
F_E &= \{((\mathbb{T}_1, \mathbb{T}_2), R(\mathbb{T}_1, \mathbb{T}_2)), ((\mathbb{T}_1, \mathbb{T}_3), R(\mathbb{T}_1, \mathbb{T}_3)), \\
& \quad ((\mathbb{T}_2, \mathbb{T}_3), R(\mathbb{T}_2, \mathbb{T}_3))\} \\
\} &
\end{aligned}
$$

This forms a complete graph, and therefore has $r = \frac{t(t-1)}{2}$ edges for t attribute types.

Since attribute types are fully labeled, corresponding relations are computed from between their types, the general NP-hard graph matching problem can be avoided. For more information, refer to previous work [159].

In our matching strategy, we are first taking the simplifying assumption that V^q and V^d are identical. We have observed that, for two graphs: G^q and G^d, their corresponding vertices, V^q and V^d, contain the same vocabulary elements. This means that one can set up a bijective matching functions $\varphi : V^q \rightarrow V^d$ and $\sigma : E^q \rightarrow E^d$. This bijection exists such that uv is an edge in graph G^q if and only if $\varphi(u)\varphi(v)$ is an edge in graph G^d. Further, we consider that ordering is preserved over the vertices sets V^q and V^d. *I.e.* $v_1 < v_2 \Rightarrow \varphi(v_1) < \varphi(v_2)$.

Inspired from [167] (see Fig. 5.16 for a complete illustration), thanks to our fixed labeling of attribute types, corresponding R alignment is possible between the two given graphs and we can provide a matching score between the two given graphs G^q and G^d,

$$\text{Dist.}(G^q, G^d) = \sum_{r \in E} \delta\left(F_E^q(r), F_E^d(\sigma(r))\right),$$

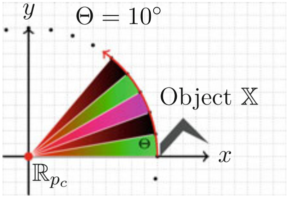
where $\delta(a, b) = ||a - b||_2$. In case where two graphs are not of exact same size, graph transformation is required (see Fig 5.16 for detailed information about vertex and edge insertion). Further, Fig. 5.17 shows a complete idea on how two labeled graphs are matched [167].

(c) *Experiments (results and comparison)*:

Let us work on a real-world industrial problem to identify a set of different known symbols in aircraft electrical wiring diagrams: FRESH dataset (see Fig. 4.3 of Chap. 4). To validate the method, a set of query images (30 queries) are taken as test samples. Note that, based on the similarity score, images are ranked in accordance with the applied query. As mentioned in Chap. 3, for evaluation, retrieval efficiency (η_K) for the short list K is used.

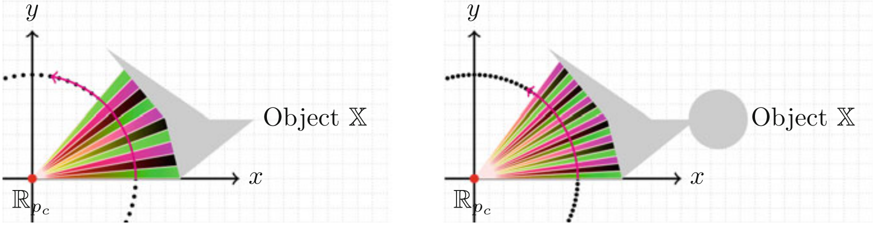
In the following, results are discussed (with comparison) and they are categorized as follows:

(a) Radial line model (see Fig. 5.18) and
(b) Basic relations (see Fig. 5.19).



Histograms:
boolean $H(\mathbb{X}, \mathbb{R}_{p_c}) =$ `[1 1 0 0 0 ... 0]`$_{1\times36}$
metric $H(\mathbb{X}, \mathbb{R}_{p_c}) =$ `[0.79 0.21 0 0 0 ... 0]`$_{1\times36}$

Fig. 5.14 Relational signature/histogram using radial line rotation



(a) $\Theta = 5°$ (b) $\Theta = 3°$

Fig. 5.15 Relational signature/histogram will be changed in accordance with the object's shape, given a reference point

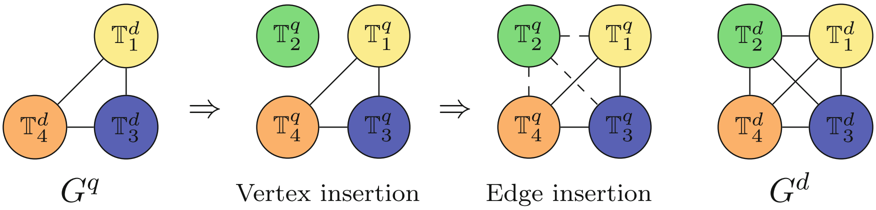
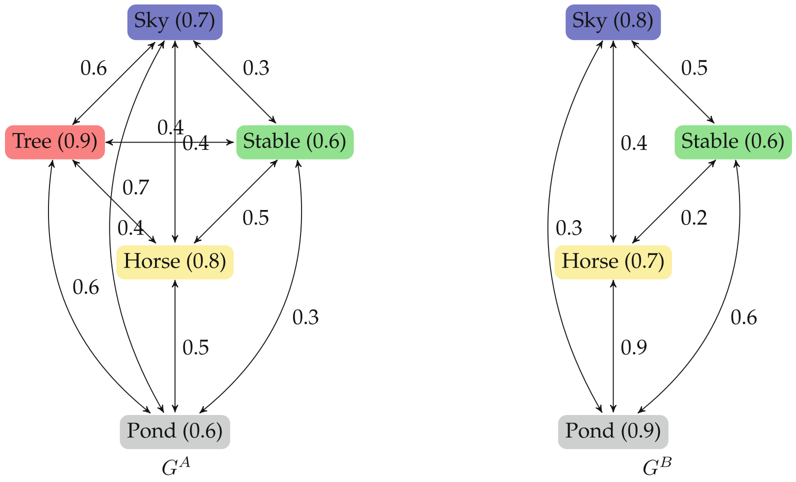


Fig. 5.16 A graph transformation: $G^q \rightarrow G^d$

In Fig. 5.18, for radial line model, a series of tests with Θ varying over $\{1°, 3°, 5°, 7°, \text{and } 9°\}$ were made. Without surprise, the lower the Θ, the better the results. In Fig. 5.19), results from cone-shaped [82], MBR [156], and angle histogram [85] are shown, where they cannot be compared with the results from radial line model.

For a comparison, in addition to shape descriptors (statistical approaches in Chap. 4), "graphical symbol recognition using spatial relations" can also be compared with other methods that are designed for graphical symbol recognition, such as statistical integration of histogram array (SIHA) [168] and kernel density matching (KDM) [169]. For detailed information can be obtained from [159].



Matching cost between G^A and G^B:

Step 1. Vertex substitution cost:	'Sky'	$\lvert 0.7 - 0.8 \rvert = 0.1$
	'Stable'	$\lvert 0.6 - 0.6 \rvert = 0.0$
	'Horse'	$\lvert 0.8 - 0.7 \rvert = 0.1$
	'Pond'	$\lvert 0.6 - 0.9 \rvert = 0.3$
Step 2. Edge substitution cost:	'Sky'↔'Pond'	$\lvert 0.4 - 0.3 \rvert = 0.1$
	'Sky'↔'Horse'	$\lvert 0.4 - 0.4 \rvert = 0.0$
	'Sky'↔'Stable'	$\lvert 0.3 - 0.5 \rvert = 0.2$
	'Horse'↔'Stable'	$\lvert 0.5 - 0.2 \rvert = 0.3$
	'Horse'↔'Pond'	$\lvert 0.5 - 0.9 \rvert = 0.4$
	'Stable'↔'Pond'	$\lvert 0.3 - 0.6 \rvert = 0.3$
Step 3. Vertex insertion cost:	'Tree' in G^A	0.9
4. Edge insertion cost:	'Tree'↔'Pond' in G^A	0.6
	'Tree'↔'Sky' in G^A	0.6
	'Tree'↔'Stable' in G^A	0.4
	'Tree'↔'Horse' in G^A	0.7

$\therefore Dist.(G^A, G^B) =$ (substitution + insertion) costs $= 5.0$.

Fig. 5.17 Computing matching cost between two graphs: $G^A \rightarrow G^B$

5.6.2 *Extension: Symbol Spotting*

In symbol spotting/localization problem (real documents) that are composed of individual parts and constrained by spatial relations, global signal-based descriptors may not be an appropriate choice since they are, unfortunately, primarily designed for applications where line symbols are isolated. In Chap. 4, we have discussed the same.

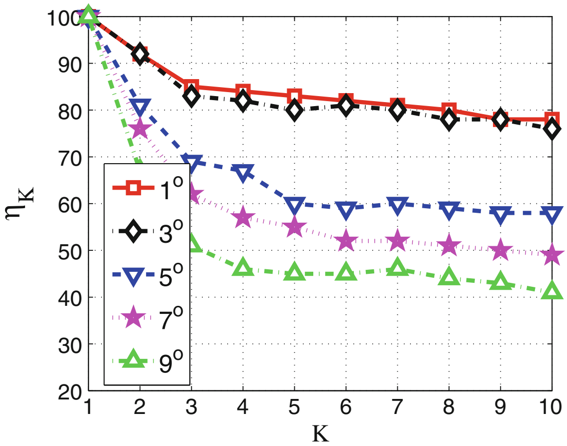


Fig. 5.18 Average retrieval efficiency (η_K) for the list $K = [1, \ldots, 10]$ using the radial line model. It uses different resolutions: 1°, 3°, 5°, 7°, and 9°. The higher the resolution (meaning less θ), the better the retrieval performance (quality)

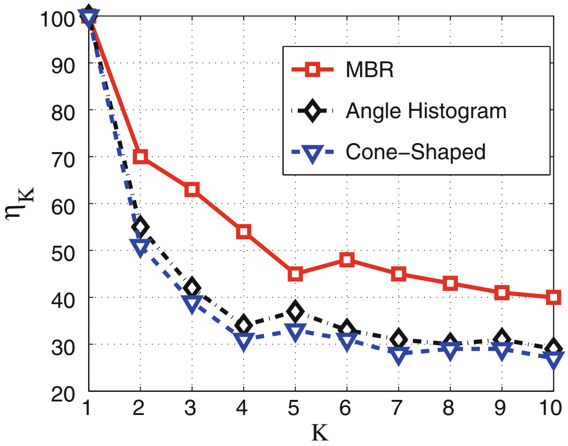


Fig. 5.19 Average retrieval efficiency (η_K) for the short list $K = [1, \ldots, 10]$ using basic spatial relations: MBR, cone-based (star calculus), and angle-based techniques

Further, if the problem is related to the segmentation/recognition paradigm [163], an accurate segmentation of meaningful parts/regions is expected. These meaningful primitives like *saliency points*, *lines*, and *arcs* can then be used to formalize the possible links that exist between them to build a graph-like structure. Graph-based symbol recognition techniques are powerful but can suffer from time complexity issues. In Sect. 5.6.1, we have just discussed, where time complexity issue has been avoided. Within the framework, can we use *Bag-of-Relations* (BoR) indexing will reduce the execution time during the symbol recognition/localization process(es).

As reported in [159], meaningful primitives are basically used for building BoRs based on their pairwise topological relations and directional relations. In other words, topological and directional relations for all possible combinations are integrated. Bags correspond to topological relations between the visual primitives, where directional relations are further computed so that relation precision can further be exploited. The idea behind the use of two different types of spatial relations is that one of the two relations cannot exploit rich information. For example, topological relation: *disconnected* does not convey any information about how visual primitives are oriented. This means that directional relations may provide additional/useful information for image recognition. As a consequence, combining them could be a better idea. Note that the number of bags is limited to the number of possible topological relations, regardless of shape, size, and number of the visual primitives that compose the symbol. In each bag, directional relations are computed and stored. Consequently, for recognition, directional relation matching takes place only with those which share similar topological and vocabulary type information. This not only simply reduces the computational complexity in matching but also avoids irrelevant relation matching, thanks to the labeled primitive type.

In this previous section, visual primitives are grouped by type. While the idea can avoid the NP-hardness of the underlying graph matching problem, it requires at least two different types of visual primitives in a symbol to compute the needed spatial relations. It is therefore inappropriate for symbols having only a single vocabulary type (e.g., four *corners* from a rectangle-shaped symbol) regardless how many visual primitives it contains. However, computing all possible relations that exist between individual visual primitives is computationally expensive but, it is possible to reduce the execution using *Bag-of-Relations* (BoR) indexing.

(a) *Bag-of-relations*:

Any symbol $\mathscr{S}$ is decomposed into a variable number p of visual primitives, each of which belongs to a vocabulary type $\mathbb{T}_t$ (in our case $1 \leq t \leq 4$). For any vocabulary type $\mathbb{T}_t$, there are m_t visual primitives,

$$\mathbb{T}_t = \{\wp_i^t\}, i = [1, \ldots, m_t] \text{ and } p = \textstyle\sum_t m_t. \tag{5.2}$$

Any pair of primitives $(\wp_1, \wp_2)$, as illustrated in Fig. 5.20, can be represented by both the vocabulary types each part belongs to (represented by their color) and by the topological relation that characterizes them: (a) *disconnected* (DC), (b) *externally connected* (EC), (c) *overlap* (O), (d) *contain/inside* (Cn/I), (e) *cover/covered by* (Cr/CB), and (f) *equal* (EQ). This means that in general, visual primitives are categorized into six topological relations. For simplicity, we rewrite such a set of topological relations as

$$\text{Categorization} = \{\mathscr{C}_{\mathbb{k}}\}, \mathbb{k} = [1, \ldots, 6], \tag{5.3}$$

preserving the label ordering $\{\mathscr{C}_{\text{DC}}, \mathscr{C}_{\text{EC}}, \mathscr{C}_{\text{O}}, \ldots, \mathscr{C}_{\text{EQ}}\}$.

To obtain the topological relation between two primitives $T(\wp_1, \wp_2)$, we use the 9-intersection model [154, 170, 171] relative to the boundaries $(\partial *)$, interiors $(*^o)$ and exteriors $(*^-)$ of $\wp_1$ and $\wp_2$:

$$T(\wp_1, \wp_2) = \begin{bmatrix} \wp_1^o \cap \wp_2^o & \wp_1^o \cap \partial\wp_2 & \wp_1^o \cap \wp_2^- \\ \partial\wp_1 \cap \wp_2^o & \partial\wp_1 \cap \partial\wp_2 & \partial\wp_1 \cap \wp_2^- \\ \wp_1^- \cap \wp_2^o & \wp_1^- \cap \partial\wp_2 & \wp_1^- \cap \wp_2^- \end{bmatrix}. \tag{5.4}$$

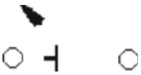
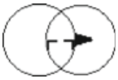
Their definitions use basic set operations like $=$, $\neq$, $\subseteq$ and $\cap$ [170]. For example,

- *equal*$(\wp_1, \wp_2)$:= points$(\wp_1)$ = points$(\wp_2)$,
- *disconnected*$(\wp_1, \wp_2)$:= points$(\wp_1) \neq$ points$(\wp_2)$ or points$(\wp_1) \cap$ points$(\wp_2)$ $= \emptyset$,
- *inside*$(\wp_1, \wp_2)$:= points$(\wp_1) \subseteq$ points$(\wp_2)$, and
- *intersects*$(\wp_1, \wp_2)$:= points$(\wp_1) \cap$ points$(\wp_2) \neq \emptyset$.

Since the *intersects* definition covers both *equal* and *inside*, they must be separated. Therefore, the previous definitions have been augmented with the consideration of boundary and interior so that the *overlap* and *externally connected* can be distinguished [172]:

- *overlap*$(\wp_1, \wp_2) := \partial\wp_1 \cap \partial\wp_2 \neq \emptyset \;\&\; \wp_1^o \cap \wp_2^o \neq \emptyset$; and
- *externally connected*$(\wp_1, \wp_2) := \partial\wp_1 \cap \partial\wp_2 \neq \emptyset \;\&\; \wp_1^o \cap \wp_2^o = \emptyset$.

Therefore, the topological relation $T(\wp_1, \wp_2)$ provides a Boolean value for each of the elements of the matrix shown in (5.4). It is straightforward to combine these elements to obtain $\{\mathscr{C}_{\Bbbk}\}$.
Following visual primitives in Fig. 5.2, let us have a few examples that can help understand the process of how bags (topology-based) can be created.

-  (symbol 1 in Fig. 5.2):
  In this case, all possible combinations of visual primitives are found to be in *disconnected* configurations except two neighboring *corners*: *southeast* ⌟ and *northeast* ⌝. As a consequence, we have two different bags: *disconnected* and *externally connected*.
-  (symbol 2 in Fig. 5.2):

  In this example, *circles* ○ are *overlapped*. When *thick* ▶ is taken into account, two different topological relations, i.e., *externally connected* (with *circle* on the left) and *contain/inside* (with *circle* on the right) are found. Similarly, *northeast* corner ⌝ is *externally connected* with *thick*, *northwest* corner ⌜ is *disconnected* with *thick*, and corners are *disconnected*. Both corners are *covered by* circles. On the whole, we have four bags: *overlap*, *externally connected*, *cover/covered by*, and *contain/inside*.

Note that, in order to avoid computing several relations, visual primitive: *extremity* is not taken into account. For all of these pairs (in every bag), corresponding directional relations are computed using the radial line model (as illustrated in

Fig. 5.14). In Fig. 5.21, two relational signatures (histograms) are shown.

In the matching process, we can only consider bags where matching candidate pairs share identical values for their vocabulary types and topological relations, thanks to the indexing of topological bags, i.e., bag-of-relations. Further, as an example, Fig. 5.22 shows an example, where matching can be significantly reduced since unlike the conventional matching procedure, directional relation matching happens only when pairs (of visual primitives) share exactly similar vocabulary type information with the query pair in that particular bag.

(b) *Results and analysis*:

Like the previous concept (mentioned in Sect. 5.5), it may not be an appropriate choice for isolated graphical symbol recognition, especially in case of degradation, noise, and deformations. In those latter cases, one cannot extract expected visual primitives and therefore spatial relations are missed. However, for the clean data, the method performs as best as other state-of-the-art graphics recognition methods.

On the other side, the interesting part of the work is, it can be used for all kinds of graphical symbol/element spotting in cases: composite and complex electrical and architectural circuit diagrams.

(a) For the FRESH dataset (as mentioned before: Fig. 4.3 of Chap. 4), using the exact same evaluation protocol (metric), average retrieval efficiency (η_K) for the short list $K = [1, \ldots, 10]$ is shown Fig. 5.23. The results can be compared with Fig. 5.18, where vanilla or straight radial line model is employed. BoRs (Fig. 5.23) provide a performance improvement by approximately 3–4% compared to Fig. 5.18.

Additional interesting comparison can be summarized as follows. It is important to comment on the previous approach [159] that is relying on an ARG framework, which requires minimum two visual primitive types. As a consequence, isolated symbols (with a single visual primitive type) cannot be handled. For example, in the GREC dataset, a symbol can sometimes be composed of a collection of a single vocabulary type: four *corners* for any rectangle-shaped symbol. This is how the method is different from other work, reported earlier [159].

(b) Another similar experimental test can prove the effectiveness of the BORs concept. For this, let us consider SESYD datasets[1] [173]. It contains two different datasets: (i) bag-of-symbols (BoS) and (ii) electrical diagrams (circuit) (ED).

In both datasets, a model symbol is used as a query to retrieve and or spot the exactly similar (or similar) symbols from all database. From each database image, multiple similar symbols are expected to be retrieved. Note that, as before, since the ground truth varies from one query symbol to another, retrieval efficiency (for evaluation) is computed. In Fig. 5.24, two examples (one per dataset) illustrating symbol retrieval using (a) three different queries in BoS dataset and (b) four different queries in ED dataset.

[1] http://mathieu.delalandre.free.fr/projects/sesyd/.

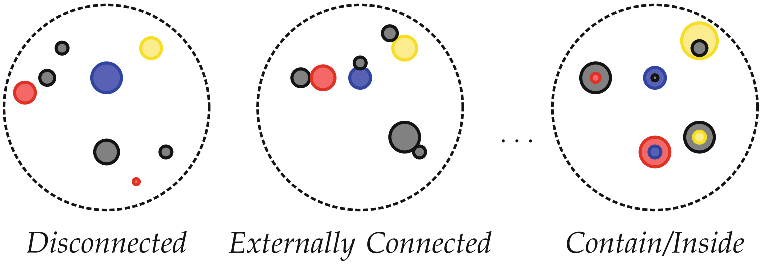


Fig. 5.20 Bag-of-relations (BoRs) model: each item in every bag represents a visual primitive and its color represents its vocabulary type

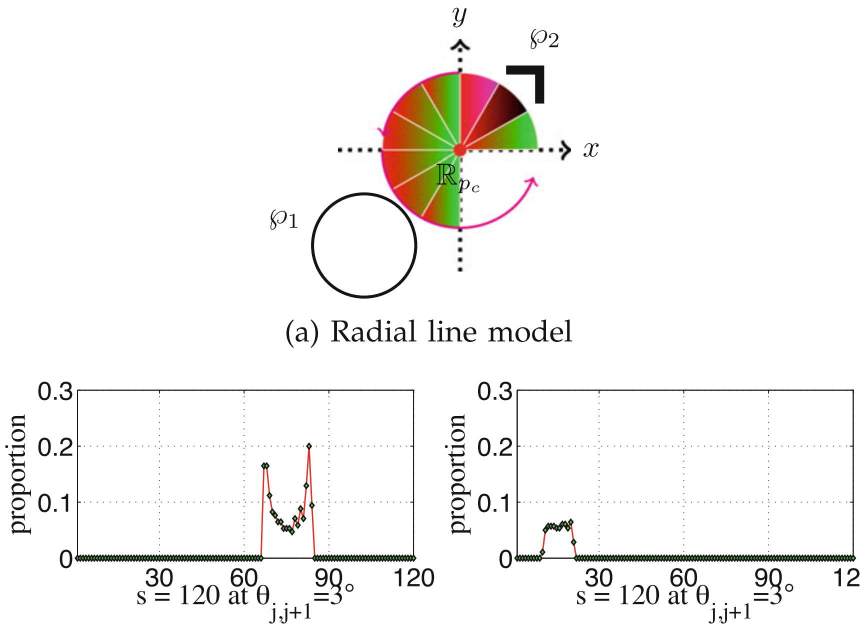
(c) *Can it be user-friendly?*
Following the idea of how it works, we observe that it could be used as a user-friendly tool for graphical symbol/element spotting (retrieval). In a few words, user takes a set of visual primitives. Since user provides their queries by selecting pair(s) of visual primitive, a user-friendly symbol retrieval test can be made. Following Fig. 5.25, two queries (explained below) can be considered to check whether the concept/technique can be user-friendly:

Query 1 *"Retrieve symbols with a thick inside a circle"*.
Based on the query description, symbols are retrieved from the database. In this example, one can see no shape and size information about visual primitives has been taken into account.

Query 2 *"Retrieve rectangle-shaped symbols"*.
To illustrate it, we use a set of four *corners* facing each other representing a *rectangle* and retrieve database symbols accordingly.

5.7 Summary

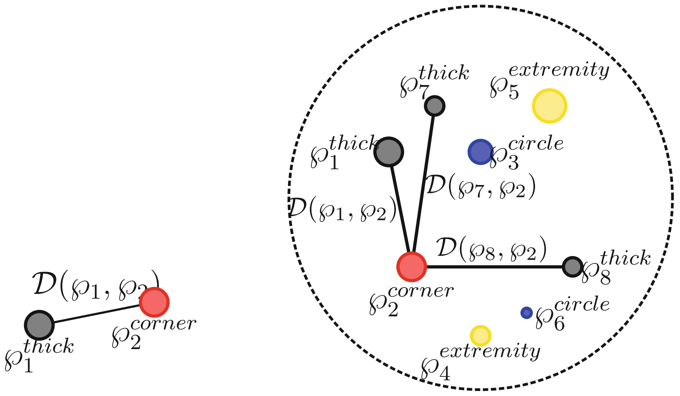
In this chapter, we have comprehensively reported several different (but, major) structural approaches for graphical symbol recognition, retrieval, and spotting. Further, usefulness of visual cues or meaning parts that compose a graphical symbol together with their spatial relations is explained. Common methods used to extract visual cues are explained and experimented on graphics recognition problem. In a similar fashion, graph-based graphical symbol recognition techniques are explained, where the use of spatial relations is primarily focused. Testing on several different datasets (graphics recognition), in this chapter, effect of spatial relations has been proved. Next chapter will discuss few possibilities to combine/integrate statistical and structural approaches for graphics recognition.



(a) Radial line model

(b) Histograms for both visual primitives: $\wp_1$ (left) and $\wp_2$ (right).

Fig. 5.21 Two disconnected pairs of primitives $\wp_1$ (circle) and $\wp_2$ (corner) and directional relational histograms using the radial line model (RLM) with respect to the unique reference point $\mathbb{R}_{p_c}$. RLM is applied for both primitives, and the unique reference point is derived using their topological relations. More information about RLM can be found in Sect. 5.6.1



(a) Query pair

(b) Database pairs (*disconnected* category)

Fig. 5.22 Matching directional relations in *disconnected* category. Each item represents a visual primitive and color represents its vocabulary type

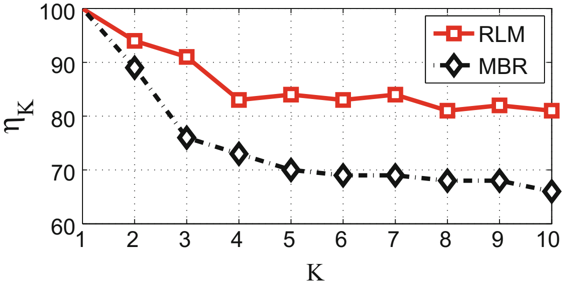


Fig. 5.23 Average retrieval efficiency (η_K) for the short list $K = [1, \ldots, 10]$ on FRESH dataset using the BORs concept

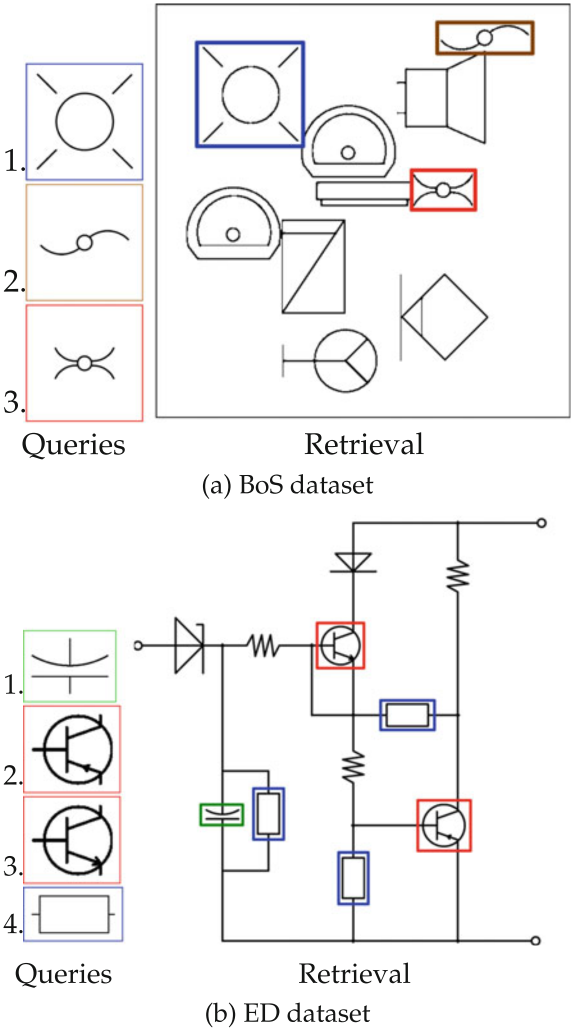


Fig. 5.24 Two examples (one per dataset) illustrating symbol retrieval using **a** three different queries in BoS dataset and **b** four different queries in ED dataset

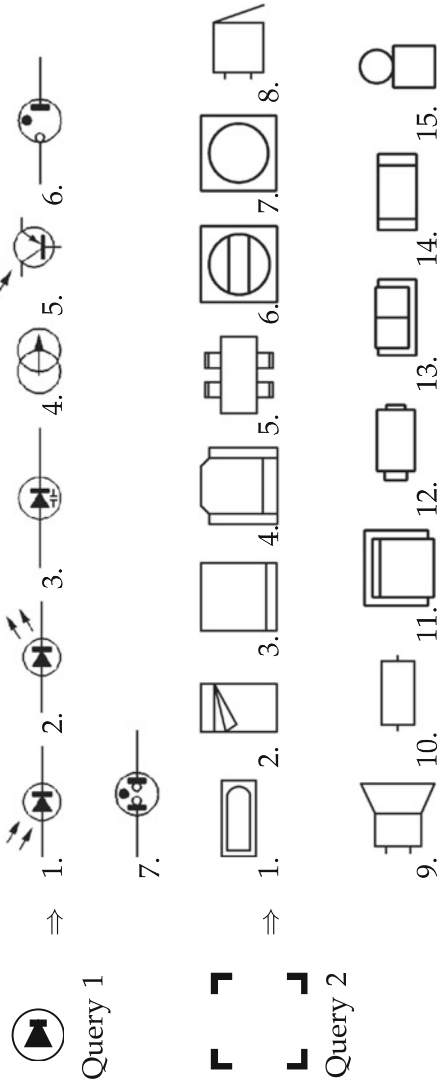


Fig. 5.25 Symbol retrieval in accordance with the user's choice: visual primitives. For this, FRESH and GREC datasets are used

References

1. W.H. Tsai, K.S. Fu, Attributed grammar: a tool for combining syntactic and statistical approaches to pattern recognition. IEEE Trans. Syst. Man Cybern. **10**(12), 873–885 (1980)
2. J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *Graphics Recognition - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 104–127
3. B.T. Messmer, H. Bunke, Automatic learning and recognition of graphical symbols, in *engineering drawings, in Graphics Recognition-Methods and Applications*, ed. by R. Kasturi,

K. Tombre, Lecture Notes, in Computer Science, vol. 1072, (Springer, Berlin, 1996), pp. 123–134

4. J.-Y. Ramel, G. Boissier, H. Emptoz, A structural representation adapted to handwritten symbol recognition, in *Proceedings of 3rd International Workshop on Graphics Recognition, Jaipur (India)* (1999), pp. 259–266
5. K.C. Santosh, *Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis)*. Ph.D. thesis, University of Lorraine, France (2011)
6. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: Bag-of-relations for symbol retrieval. Int. J. Pattern Recognit Artif Intell. **28**(06), 1450017 (2014)
7. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, 2015), pp. 1–22
8. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi. Communications in Computer and Information. Science **709**, 3–15 (2017)
9. K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005)
10. D.S. Doermann, An introduction to vectorization and segmentation, in *Graphics Recognition—Algorithms and Systems*, ed. by K. Tombre, A.K. Chhabra. *Lecture Notes in Computer Science*, vol. 1389 (Springer, 1998), pp. 1–8
11. J.Y. Chiang, S.C. Tue, Y.C. Leu, A new algorithm for line image vectorization. Pattern Recogn. **31**(10), 1541–1549 (1998)
12. Y. Zheng, H. Li, D. Doermann, A parallel-line detection algorithm based on HMM decoding. IEEE Trans. Pattern Anal. Mach. Intell. **27**(5), 777–792 (2005)
13. D. Dori, Vector-based arc segmentation in the machine drawing understanding system environment. IEEE Trans. Pattern Anal. Mach. Intell. **17**(11), 1057–1068 (1995)
14. Ph. Dosch, G. Masini, K. Tombre, Improving arc detection in graphics recognition, in *Proceedings of the 15th International Conference on Pattern Recognition, Barcelona (Spain)*, vol. 2 (2000), pp. 243–246
15. B. Lamiroy, Y. Guebbas, Robust and precise circular arc detection, in *Graphics Recognition. Achievements, Challenges, and Evolution, 8th International Workshop, GREC 2009, La Rochelle, France, July 22-23, 2009. Selected Papers*, ed. by J.-M. Ogier, L. Wenyin, J. Lladós. *Lecture Notes in Computer Science*, vol. 6020 (Springer, 2010), pp. 49–60
16. R. Kasturi, S. Bow, J. Gattiker, J. Shah, W. El-Masri, U. Mokate, S. Honnenahalli, A system for recognition and description of graphics, in *Proceedings of 9th International Conference on Pattern Recognition, Rome (Italy)* (1988), pp. 255–259
17. C.C. Shih, R. Kasturi, Extraction of graphical primitives from images of paper based line drawings. Mach. Vis. Appl. **2**, 103–113 (1989)
18. D. Lysak, R. Kasturi, Interpretation of Line Drawings with Multiple Views **1**, 220–222 (1990)
19. P. Kultanen, E. Oja, L. Xu, Randomized Hough Transform (RHT) in engineering drawing vectorization system, in *Proceedings of IAPR Workshop on Machine Vision Applications, Tokyo (Japan)* (1990), pp. 173–176
20. D. Dori, Orthogonal zig-zag: an algorithm for vectorizing engineering drawings compared with hough transform. Adv. Eng. Softw. **28**(1), 11–24 (1997)
21. L. Lam, S.-W. Lee, C.Y. Suen, Thinning methodologies - a comprehensive survey. IEEE Trans. Pattern Anal. Mach. Intell. **14**(9), 869–885 (1992)
22. G. Sanniti di Baja, Well-shaped, stable, and reversible skeletons from the (3,4)-distance transform. J. Vis. Commun. Image Represent. **5**(1), 107–115 (1994)
23. C.S. Fahn, J.F. Wang, J.Y. Lee, A topology-based component extractor for understanding electronic circuit diagrams. Comput. Vis. Graph. Image Process. **44**, 119–138 (1988)
24. R. Kasturi, S.T. Bow, W. El-Masri, J. Shah, J.R. Gattiker, U.B. Mokate, A system for interpretation of line drawings. IEEE Trans. Pattern Anal. Mach. Intell. **12**(10), 978–992 (1990)
25. R.D.T. Janssen, A.M. Vossepoel, Adaptive vectorization of line drawing images. Comput. Vis. Image Underst. **65**(1), 38–56 (1997)

26. X. Hilaire, K. Tombre, Robust and accurate vectorization of line drawings. IEEE Trans. Pattern Anal. Mach. Intell. **28**(6), 890–904 (2006)
27. D. Antoine, S. Collin, K. Tombre, Analysis of technical documents: the REDRAW system, in *Pre-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Murray Hill, NJ (USA)* (1990), pp. 1–20
28. I. Chai, D. Dori, Orthogonal zig-zag: an efficient method for extracting lines from engineering drawings, in *Visual Form*, ed. by C. Arcelli, L.P. Cordella, G. Sanniti di Baja (Plenum Press, New York and London, 1992), pp. 127–136
29. D. Dori, W. Liu, Sparse pixel vectorization: an algorithm and its performance evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **21**(3), 202–215 (1999)
30. T.J. Davis, Fast decomposition of digital curves into polygons using the haar transform. IEEE Trans. Pattern Anal. Mach. Intell. **21**(8), 786–790 (1999)
31. P.L. Rosin, Techniques for assessing polygonal approximation of curves. IEEE Trans. Pattern Anal. Mach. Intell. **19**(6), 659–666 (1997)
32. P.Y. Yin, A new method of polygonal approximation using genetic algorithm. Pattern Recogn. Lett. **19**, 1017–1026 (1998)
33. P.Y. Yin, A tabu search approach to polygonal approximation of digital curves. Int. J. Pattern Recognit Artif Intell. **14**(2), 243–255 (2000)
34. U. Ramer, An iterative procedure for the polygonal approximation of plane curves. Comput. Graph. Image Process. **1**, 244–256 (1972)
35. K. Wall, P. Danielsson, A fast sequential method for polygonal approximation of digitized curves. Comput. Vis. Graph. Image Process. **28**, 220–227 (1984)
36. J. Sklansky, V. Gonzalez, Fast polygonal approximation of digitized curves. Pattern Recogn. **12**, 327–331 (1980)
37. J.C. Perez, E. Vidal, Optimum polygonal approximation of digitized curves. Pattern Recogn. Lett. **15**(8), 743–750 (1994)
38. A. Kolesnikov, P. Fränti, Data reduction of large vector graphics. Pattern Recogn. **38**, 381–394 (2005)
39. M. Salotti, An efficient algorithm for the optimal polygonal approximation of digitized curves. Pattern Recogn. Lett. **22**(2), 215–221 (2001)
40. P.L. Rosin, G.A. West, Segmentation of edges into lines and arcs. Image Vis. Comput. **7**(2), 109–114 (1989)
41. C.-H. Teh, R.T. Chin, On the detection of dominant points on digital curves. IEEE Trans. Pattern Anal. Mach. Intell. **11**(8), 859–872 (1989)
42. W.-Y. Wu, M.-J.J. Wang, Detecting the dominant points by the curvature-based polygonal approximation **55**, 79–88 (1993)
43. N. Ansari, K.W. Huang, Non-parametric dominant point detection. Pattern Recogn. **24**(9), 849–862 (1991)
44. J.-P. Salmon, L. Wendling, ARG based on arcs and segments to improve the symbol recognition by genetic algorithm, in *Graphics Recognition. Recent Advances and New Opportunities*, ed. by W. Liu, J. Lladós, J.-M. Ogier. *Lecture Notes in Computer Science*, vol. 5046 (Springer, 2007), pp. 80–90
45. D. Elliman, Tif2vec, an algorithm for arc segmentation in engineering drawings, in *Graphics Recognition Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon. *Lecture Notes in Computer Science*, vol. 2390 (Springer, 2002), pages 350–358
46. R.S. Conker, A dual plane variation of the hough transform for detecting non-concentric circles of different radii. Comput. Vis. Graph. Image Process. **43**, 115–132 (1988)
47. V.F. Leavers, The dynamic generalized hough transform: its relationship to the probabilistic hough transforms and an application to the concurrent detection of circles and ellipses. CVGIP **56**(3), 381–398 (1992)
48. W. Liu, D. Dori, Incremental arc segmentation algorithm and its evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **20**(4), 424–431 (1998)
49. J. Song, M.R. Lyu, S. Cai, Effective multiresolution arc segmentation: algorithms and performance evaluation. IEEE Trans. Pattern Anal. Mach. Intell. **26**(11), 1491–1506 (2004)

50. L.P. Cordella, M. Vento, Symbol recognition in documents: a collection of techniques? Int. J. Doc. Anal. Recogn. **3**(2), 73–88 (2000)
51. K. Tombre, D. Dori, Interpretation of engineering drawings, in *Handbook of Character Recognition and Document Image Analysis*, ed. by H. Bunke, P.S.P. Wang. chapter 17 (World Scientific, 1997), pp. 457–484
52. K. Tombre, Analysis of engineering drawings: state of the art and challenges, in *Proceedings of 2nd International Workshop on Graphics Recognition, Nancy (France)* (1997), pp. 54–61
53. L. Wenyin, J. Zhai, D. Dori,. Extended summary of the arc segmentation contest, in *Graphics Recognition Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon. *Lecture Notes in Computer Science*, volume 2390 (Springer, 2002), pp. 343–349
54. X. Hilaire, RANVEC and the arc segmentation contest, in *Graphics Recognition – Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon. *Lecture Notes in Computer Science*, vol. 2390 (Springer, 2002), pp. 359–364
55. L. Wenyin, Report of the arc segmentation contest, in *Graphics Recognition, Recent Advances and Perspectives*, ed. by J. Lladós, Y.-B. Kwon. *Lecture Notes in Computer Science*, vol. 3088 (Springer, 2004), pp. 364–367
56. M. Tooley, D. Wyatt, *Aircraft engineering principles and practice* (Principles, Operation and Maintenance (Butterworth-Heinemann, Aircraft Electrical and Electronic Systems, 2008)
57. P.M. Devaux, D.B. Lysak, R. Kasturi, A complete system for the intelligent interpretation of engineering drawings. Int. J. Doc. Anal. Recogn. **2**(2/3), 120–131 (1999)
58. P. Dosch, K. Tombre, C. Ah-Soon, G. Masini, A complete system for analysis of architectural drawings. Int. J. Doc. Anal. Recogn. **3**(2), 102–116 (2000)
59. J. Rendek, G. Masini, Ph. Dosch, K. Tombre, The search for genericity in graphics recognition applications: design issues of the Qgar software system, in *Proceedings of the 6th IAPR International Workshop on Document Analysis Systems, Florence, (Italy). Lecture Notes in Computer Science*, vol. 3163 (2004), pp. 366–377
60. H.S.M. Al-Khaffaf, A. Zawawi Talib, M. Azam Osman, Final report of grec'11 arc segmentation contest: performance evaluation on multi-resolution scanned documents, in *Proceedings of IAPR International Workshop on Graphics Recognition*, ed. by Y.-B. Kwon, J.-M. Ogier. *Lecture Notes in Computer Science*, vol. 7423 (Springer, 2013), pp. 187–197
61. W. Jianping, K. Chen, X. Gao, Fast and accurate circle detection using gradient-direction-based segmentation. J. Opt. Soc. Am. A **30**(6), 1184–1192 (2013)
62. R.O. Duda, P. Hart, Use of the hough transformation to detect lines and curves in pictures. Commun. ACM **15**(1), 11–15 (1972)
63. X. Lei, E. Oja, P. Kultanen, A new curve detection method: randomized hough transform (rht). Pattern Recogn. Lett. **11**(5), 331–338 (1990)
64. A. Ajdari Rad, K. Faez, N. Qaragozlou, Fast circle detection using gradient pair vectors, in *Proceedings of the Seventh International Conference on Digital Image Computing: Techniques and Applications*, ed. by C. Sun, H. Talbot, S. Ourselin, T. Adriaansen (CSIRO Publishing, 2003), pp. 879–888
65. K. Chen, W. Jianping, One-dimensional voting scheme for circle and arc detection. J. Opt. Soc. Am. A **31**(12), 2593–2602 (2014)
66. S. Saqib Bukhari, H.S.M. Al-Khaffaf, F. Shafait, M. Azam Osman, A. Zawawi Talib, T.M. Breuel, Final report of grec'13 arc and line segmentation contest, in *Graphics Recognition. Current Trends and Challenges*, ed. by B. Lamiroy, J.-M. Ogier. *Lecture Notes in Computer Science*, vol. 8746 (Springer, 2014), pp. 234–239
67. J. Song, F. Su, C.-L. Tai, S. Cai, An object-oriented progressive-simplification based vectorization system for engineering drawings: model, algorithm, and performance. IEEE Trans. Pattern Anal. Mach. Intell. **24**(8), 1048–1060 (2002)
68. G. Retz-Schmidt, Various Views on Spatial Prepositions. *AI Magazine* (1988), pp. 95–104
69. M. Bar, S. Ullman, Spatial context in recognition. Perception **25**, 324–352 (1993)
70. I. Biederman, Perceiving real-world scenes. Science **177**(43), 77–80 (1972)
71. C.B. Cave, S.M. Kosslyn, The role of parts and spatial relations in object identification. Perception **22**(2), 229–248 (1993)

72. J.H. Vandenbrande, A.A.G. Requicha, Spatial reasoning for the automatic recognition of machinable features in solid models. IEEE Trans. Pattern Anal. Mach. Intell. **15**(12), 1269–1285 (1993)
73. J. Silva Centeno, Segmentation of thematic maps using colour and spatial attributes, in *GREC* (1997), pp. 233–239
74. T. Gevers, A.W.M. Smeulders, Σnigma: an image retrieval system. Proc. IAPR Int. Conf. Pattern Recognit. **2**, 697–700 (1992)
75. S.-H. Lee, F.-J. Hsu, Spatial reasoning and similarity retrieval of images using 2D C-string knowledge representation. Pattern Recogn. **25**(3), 305–318 (1992)
76. G. Heidemann, Combining spatial and colour information for content based image retrieval. Comput. Vis. Image Underst. **94**, 234–270 (2004)
77. S. Medasani, R. Krishnapuram, A fuzzy approach to content-based image retrieval, in *Proceedings of FUZZ-IEEE* (1997), pp. 1251–1260
78. P.H. Winston, *The Psychology of Computer Vision* (McGraw-Hill, New York, 1975)
79. J. Freeman, The modelling of spatial relations. Comput. Graph. Image Process. **4**, 156–171 (1975)
80. J. Renz, B. Nebel, Spatial reasoning with topological information, in *An Interdisciplinary Approach to Representing and Processing Spatial Knowledge (Springer*, ed. by Spatial Cognition (UK, London, 1998), pp. 351–372
81. M.F. Worboys, *GIS - A Computing Perspective* (Taylor and Francis, 1995)
82. K. Miyajima, A. Ralescu, Spatial organization in 2D segmented images: representation and recognition of primitive spatial relations. Fuzzy Sets Syst. **2**(65), 225–236 (1994)
83. D. Mitra, A class of star-algebras for point-based qualitative reasoning in two-dimensional space, in *Fifteenth International Florida Artificial Intelligence Research Society Conference* (2002), pp. 486–491
84. J. Renz, D. Mitra, Qualitative direction calculi with arbitrary granularity, in *Proceedings of the Pacific Rim International Conferences on Artificial Intelligence* (2004), pp. 65–74
85. X. Wang, J.M. Keller, Human-based spatial relationship generalization through neural/fuzzy approaches. Fuzzy Sets Syst. **101**, 5–20 (1999)
86. E. Jungert, Qualitative spatial reasoning for determination of object relations using symbolic interval projections, in *IEEE Symposium on Visual Languages* (1993), pp. 24–27
87. R.K. Goyal, M.J. Egenhofer, Similarity of cardinal directions, in Advances in Spatial and Temporal Databases. Lecture Notes in Computer Science **2121**, 36–55 (2001)
88. S. Dutta, Approximate spatial reasoning: integrating qualitative and quantitative constraints. Int. J. Approx. Reason. **5**, 307–331 (1991)
89. S.M.R. Dehak, *Inference Quantitative des Relations Spatiales Directionnelles*. Ph.d. thesis, École Nationale Supérieure des Télécommunications (2002)
90. L.A. Zadeh, Fuzzy sets. Inf. Control **8**, 338–353 (1965)
91. M.J. Egenhofer, A. Rashid, Shariff, Metric details for natural-language spatial relations. ACM Trans. Inf. Syst. **16**(4), 295–321 (1998)
92. I. Bloch, Fuzzy relative position between objects in image processing: a morphological approach. IEEE Trans. Pattern Anal. Mach. Intell. **21**(7), 657–664 (1999)
93. P. Matsakis, L. Wendling, A new way to represent the relative position between areal objects. IEEE Trans. Pattern Anal. Mach. Intell. **21**(7), 634–643 (1999)
94. A. Morris, A framework for modeling uncertainty in spatial databases. Trans. GIS **7**, 83–101 (2003)
95. H. Bunke, K. Riesen, Recent advances in graph-based pattern recognition with applications in document analysis. Pattern Recogn. **44**(5), 1057–1067 (2011)
96. L.R. Foulds, *Graph Theory Applications*. Universitext (1979) (Springer, New York, 1992)
97. J. Lladós, G. Sánchez, Graph matching versus graph parsing in graphics recognition - a combined approach. Int. J. Pattern Recognit Artif Intell. **18**(3), 455–473 (2004)
98. A. Robles-Kelly, E.R. Hancock, String edit distance, random walks and graph matching. Int. J. Pattern Recognit. Artif. Intell. **18**(03), 315–327 (2004)

99. H. Bunke, B.T. Messmer, Recent advances in graph matching. Int. J. Pattern Recognit. Artif. Intell. **11**(01), 169–203 (1997)
100. P. Foggia, G. Percannella, M. Vento, Graph matching and learning in pattern recognition in the last 10 years. Int. J. Pattern Recognit. Artif. Intell. **28**(1), 1450001 (2014)
101. H.P. Morevec, Towards automatic visual obstacle avoidance, in *Proceedings of International Joint Conference on Artificial Intelligence* (1977), pp. 584–584
102. D.G. Lowe, Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**(2), 91–110 (2004)
103. M. Rusiñol, J. Lladós, Word and symbol spotting using spatial organization of local descriptors, in *Proceedings of International Workshop on Document Analysis Systems* (2008), pp. 489–496
104. K. Mikolajczyk, C. Schmid, Scale and affine invariant interest point detectors. Int. J. Comput. Vision **60**(1), 63–86 (2004)
105. A. Rosenfeld, Adjacency in digital pictures. Inf. Control **26**(1), 24–33 (1974)
106. F.C.A. Groen, A.C. Sanderson, J.F. Schlag, Symbol recognition in electrical diagrams using probabilistic graph matching. Pattern Recogn. Lett. **3**, 343–350 (1985)
107. S.W. Lee, J.H. Kim, F.C.A. Groen, Translation- rotation- and scale invariant recognition of hand-drawn symbols in schematic diagrams. Int. J. Pattern Recognit. Artif. Intell. **4**(1), 1–25 (1990)
108. H. Bunke, B.T. Messmer, Efficient attributed graph matching and its application to image analysis, in *Proceedings of 8th International Conference on Image Analysis and Processing, San Remo (Italy)*, ed. by C. Braccini, L. De Floriani, G. Vernazza. *Lecture Notes in Computer Science*, vol. 974 (1995), pp. 45–55
109. B. Gun Park, K. Mu Lee, S. Uk Lee, J. Hak Lee, Recognition of partially occluded objects using probabilistic ARG (attributed relational graph)-based matching. Comput. Vis. Image Underst. **90**(3), 217–241 (2003)
110. D. Conte, P. Foggia, C. Sansone, M. Vento, Thirty years of graph matching in pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **18**(3), 265–298 (2004)
111. J. Lladós, J. López-Krahe, E. Martí, A system to understand hand-drawn floor plans using subgraph isomorphism and hough transform. Mach. Vis. Appl. **10**(3), 150–158 (1997)
112. J. Lladós, E. Martí, J.J. Villanueva, Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. IEEE Trans. Pattern Anal. Mach. Intell. **23**(10), 1137–1143 (2001)
113. C. Ah-Soon, K. Tombre, Architectural symbol recognition using a network of constraints. Pattern Recogn. Lett. **22**(2), 231–248 (2001)
114. E. Valveny, E. Martí, A model for image generation and symbol recognition through the deformation of lineal shapes. Pattern Recogn. Lett. **24**(15), 2857–2867 (2003)
115. M. Delalandre, E. Valveny, J. Lladós, Performance evaluation of symbol recognition and spotting systems: an overview, in *Proceedings of International Workshop on Document Analysis Systems*, ed. by K. Kise, H. Sako (IEEE Computer Society, 2008), pp. 497–505
116. S. Jouili, S. Tabbone, Towards performance evaluation of graph-based representation, in *Proceedings of the IAPR Graph-Based Representations in Pattern Recognition* (2011), pp. 72–81
117. S. Jouili, S. Tabbone, Hypergraph-based image retrieval for graph-based representation. Pattern Recogn. **45**(11), 4054–4068 (2012)
118. K. Tombre, S. Tabbone, Ph. Dosch, Musings on symbol recognition, in *Proceedings of 6th IAPR International Workshop on Graphics Recognition, Hong Kong* (2005), pp. 23–34
119. A.T. Berztiss, A backtrack procedure for isomorphism of directed graphs. J. ACM **20**(3), 365–377 (1973)
120. J.R. Ullmann, An algorithm for subgraph isomorphism. J. ACM **23**(1), 31–42 (1976)
121. J.L. Balcazar, J. Diaz, J. Gabarro, *Structural Complexity II, EATCS Monographs on Theorical Computer Science* (Springer, Berlin, 1990)
122. L. Burak Kara, T.F. Stahovich, An image-based, trainable symbol recognizer for hand-drawn sketches. Comput. Graph. **29**(4), 501–517 (2005)

123. W.S. Lee, L. Burak Kara, T.F. Stahovich, An efficient graph-based recognizer for hand-drawn symbols. Comput. Graph. **31**(4), 554–567 (2007)
124. B.T. Messmer, H. Bunke, Efficient subgraph isomorphism detection: a decomposition approach. IEEE Trans. Knowl. Data Eng. **12**(2), 307–323 (2000)
125. X. Xiaogang, S. Zhengxing, P. Binbin, J. Xiangyu, L. Wenyin, An online composite graphics recognition approach based on matching of spatial relation graphs. Int. J. Doc. Anal. Recogn. **7**(1), 44–55 (2004)
126. L. Wenyin, W. Qian, X. Jin, Smart sketchpad - an on-line graphics recognition system, in *Proceedings of the 6th International Conference on Document Analysis and Recognition, Seattle, WA (USA)* (2001), pp. 1050–1054
127. Y. Liu, L. Wenyin, C. Jiang, A structural approach to recognizing incomplete graphic objects, in *Proceedings of the 17th International Conference on Pattern Recognition, Cambridge (UK)* (2004), pp. 371–375
128. L.G. Shapiro, R. Haralick, Structural description and inexact matching. IEEE Trans. Pattern Anal. Mach. Intell. **3**(5), 504–519 (1981)
129. B.T. Messmer, H. Bunke, Efficient error-tolerant subgraph isomorphism detection, in *Shape, Structure and Pattern Recognition (Post-proceedings of IAPR Workshop on Syntactic and Structural Pattern Recognition, Nahariya, Israel)*, ed. by D. Dori, A. Bruckstein (World Scientific, 1995), pp. 231–240
130. B.T. Messmer, H. Bunke, A new algorithm for error-tolerant subgraph isomorphism detection. IEEE Trans. Pattern Anal. Mach. Intell. **20**(5), 493–504 (1998)
131. Ph. Dosch, J. Lladós, Vectorial signatures for symbol discrimination, in *Proceedings of 5th IAPR International Workshop on Graphics Recognition, Barcelona (Spain)* (2003), pp. 159–169
132. M. Rusiñol, J. Lladós, Symbol spotting in technical drawings using vectorial signatures, in *Proceedings of 6th IAPR International Workshop on Graphics Recognition, Hong Kong* (2005), pp. 35–45
133. L. Wenyin, W. Zhang, L. Yan, An interactive example-driven approach to graphics recognition in engineering drawings. Int. J. Doc. Anal. Recogn. **9**(1), 13–29 (2007)
134. M. Muzzamil Luqman, *Fuzzy Multilevel Graph Embedding for Recognition, Indexing and Retrieval of Graphic Document Images*. Ph.D. thesis, Francois Rabelais University of Tours France and Autonoma University of Barcelona Spain (2012)
135. M. Muzzamil Luqman, J.-Y. Ramel, J. Lladós, T. Brouard, Fuzzy multilevel graph embedding. Pattern Recognit. **46**(2), 551–565 (2013)
136. A. Dutta, J. Lladós, U. Pal, A symbol spotting approach in graphical documents by hashing serialized graphs. Pattern Recogn. **46**(3), 752–768 (2013)
137. R. Mohr, T.C. Henderson, Arc and path consistency revisited. Artif. Intell. **28**, 225–233 (1986)
138. A.H. Habacha, *Reconnaissance de symboles techniques et analyse contextuelle de schémas*. Ph.d. thesis, Institut National Polytechnique de Lorraine, Vandœuvre-lès-Nancy, June 1993
139. R.C. Wilson, E.R. Hancock, Structural matching by discrete relaxation. IEEE Trans. Pattern Anal. Mach. Intell. **19**(6), 634–648 (1997)
140. O.D. Faugeras, M. Berthod, Improving consistency and reducing ambiguity in stochastic labeling: an optimization approach. IEEE Trans. Pattern Anal. Mach. Intell. **3**, 412–423 (1981)
141. W.J. Christmas, J. Kittler, M. Petrou, Structural matching in computer vision using probabilistic relaxation. IEEE Trans. Pattern Anal. Mach. Intell. **17**(8), 749–764 (1995)
142. A. Kostin, J. Kittler, W. Christmas, Object recognition by symmetrised graph matching using relaxation labelling with an inhibitory mechanism. Pattern Recogn. Lett. **26**(3), 381–393 (2005)
143. S. Mesadini, R. Khrishnapuram, Y. Choi, Graph matching by relaxation of fuzzy assigments. IEEE Trans. Fuzzy Syst. **9**(1), 173–182 (2001)
144. R. Balasubramaniam, R. Krishnapuram, S. Medasani, S.H. Jung, M.-Y.S. Choi, Content-based image retrieval based on a fuzzy approach. IEEE Trans. Knowl. Data Eng. **16**(10), 1185–1199 (2004)

145. R.C. Wilson, E.R. Hancock, Pattern vectors from algebraic graph theory. IEEE Trans. Pattern Anal. Mach. Intell. **27**(7), 1112–1124 (2005)
146. M. Coustaty, K. Bertet, M. Visani, J.-M. Ogier, A new adaptive structural signature for symbol recognition by using a galois lattice as a classifier. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **41**(4), 1136–1148 (2011)
147. M. Visani, K. Bertet, J.-M. Ogier, Navigala: an original symbol classifier based on navigation through a galois lattice. Int. J. Pattern Recognit. Artif. Intell. **25**(04), 449–473 (2011)
148. A. Boumaiza, S. Tabbone, Symbol recognition using a galois lattice of frequent graphical patterns, in *IAPR International Workshop on Document Analysis Systems*, ed. by M. Blumenstein, U. Pal, S. Uchida (IEEE, 2012), pp. 165–169
149. A. Boumaiza, S. Tabbone, A novel approach for graphics recognition based on galois lattice and bag of words representation, in *Proceedings of International Conference on Document Analysis and Recognition* (2011), pp. 829–833
150. M. Rusiñol, J. Lladós, G. Sánchez, Symbol spotting in vectorized technical drawings through a lookup table of region strings. Pattern Anal. Appl. **13**(3), 321–331 (2010)
151. M. Rusiñol, J. Lladós, *Symbol Spotting in Digital Libraries: Focused Retrieval over Graphic-Rich Document Collections* (Springer, London, 2010)
152. P. Garnesson, G. Giraudon, Spatial context in an image analysis system, in *Proceedings of European Conference on Computer Vision* (Springer, London, UK, 1990), pp. 579–582
153. T.V. Pham, A.W.M. Smeulders, Learning spatial relations in object recognition. Pattern Recogn. Lett. **27**(14), 1673–1684 (2006)
154. M.J. Egenhofer, J.R. Herring, categorizing binary topological relations between regions, lines, and points in geographic databases, in *University of Maine, Research Report* (1991)
155. D.J. Peuquet, Z. CI-Xiang, An algorithm to determine the directional relationship between arbitrarily-shaped polygons in the plane. Pattern Recognit. **20**(1), 65–74 (1987)
156. D. Papadias, Y. Theodoridis, Spatial relations, minimum bounding rectangles, and spatial data structures. Int. J. Geogr. Inf. Sci. **11**(2), 111–138 (1997)
157. K.C. Santosh, L. Wendling, B. Lamiroy, New ways to handle spatial relations through angle plus mbr theory on raster documents, in *Proceedings of IAPR International Workshop on Graphics Recognition* (La Rochelle, France, 2009), pp. 291–302
158. K.C. Santosh, L. Wendling, B. Lamiroy, Unified pairwise spatial relations: an application to graphical symbol retrieval, in *Proceedings of IAPR International Workshop on Graphics Recognition* (2009), pp. 163–174
159. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recogn. Lett. **33**(3), 331–341 (2012)
160. J. Silva Centeno, Segmentation of thematic maps using colour and spatial attributes, in *Proceedings of 2nd International Workshop on Graphics Recognition, Nancy (France)* (1997), pp. 233–239
161. T. Gevers, A.W.M. Smeulders, Σnigma: an image retrieval system, vol. 2, pp. 697–700 (1992)
162. M. Rusiñol, A. Borràs, J. Lladós, Relational indexing of vectorial primitives for symbol spotting in line-drawing images. Pattern Recogn. Lett. **31**(3), 188–201 (2010)
163. S. Yoon, Y. Lee, G. Kim, Y. Choi, New paradigm for segmentation and recognition of handwritten numeral string, in *Proceedings of International Conference on Document Analysis and Recognition* (2001), pp. 205–209
164. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, ed. by Y.-B. Kwon, J.-M. Ogier. *Lecture Notes in Computer Science*, vol. 7423 (Springer, 2011), pp. 228–237
165. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recogn. **17**(1), 61–78 (2014)
166. K.C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2009), pp. 1330–1334
167. S. Aksoy, Spatial relationship models for image information mining (2009)

168. S. Yang, Symbol recognition via statistical integration of pixel-level constraint histograms: a new descriptor. IEEE Trans. Pattern Anal. Mach. Intell. **27**(2), 278–281 (2005)

169. W. Zhang, L. Wenyin, K. Zhang, Symbol recognition with kernel density matching. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2020–2024 (2006)

170. R. Hartmut Güting, Geo-relational algebra: A model and query language for geometric database systems, in *Proceedings of the International Conference on Extending Database Technology: Advances in Database Technology* (1988), pp. 506–527

171. M.J. Egenhofer, R. Franzosa, Point-set topological spatial relations. Int. J. Geogr. Inf. Syst. **5**(2), 161–174 (1991)

172. D. Pullar, M.J. Egenhofer, Towards formal definitions of topological relations among spatial objects, in *The Third International Symposium on Spatial Data Handling*, ed. by D. Marble (1988), pp. 225–242

173. M. Delalandre, E. Valveny, T. Pridmore, D. Karatzas, Generation of synthetic documents for performance evaluation of symbol recognition and spotting systems. Int. J. Doc. Anal. Recogn. **13**(3), 187–207 (2010)

Chapter 6
Hybrid Approaches

6.1 Context

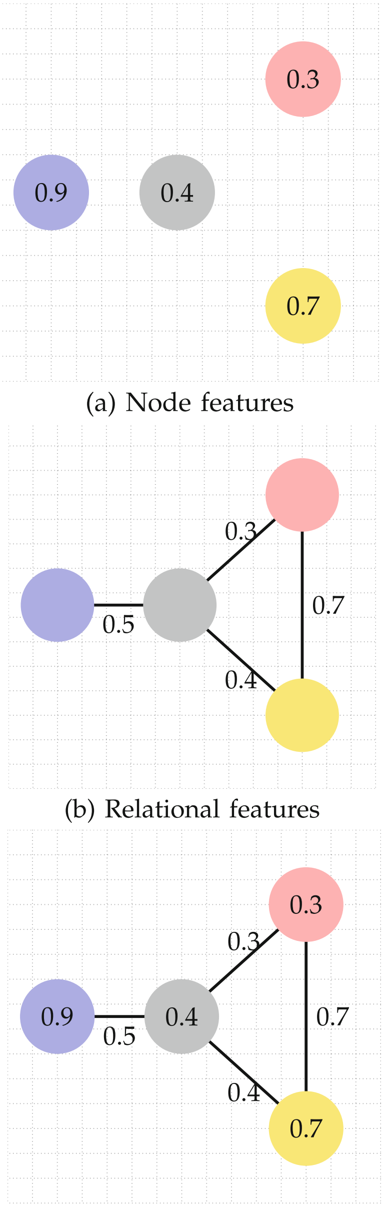
To effectively work on graphical symbol localization in real documents, one is able to identify meaningful parts (regions-of-interest) that can help characterize their shape description and formalize the possible links that exist between them. This means that integrating spatial relations (edge feature in a graph) and shape description of the extracted visual parts/primitives (as a node feature in a graph) can help enrich the graphical symbol description [1].

Not a surprising (refer to Chap. 5), in the literature, structural approaches are found to be powerful representations since they can represent how individual parts (regions-of-interest) are connected (in the form of graph). However, relations, as discussed in Chap. 5, always do not exploit shape information. In addition, in case when we need to work on local graphical elements that can exhibit subtle differences, global signal-based shape descriptors may not be a good choice. In order to exploit complete information, we are required to describe local shapes.

In this chapter, let us try and combine both structural and statistical approaches, and try to avoid the shortcomings of each of them. For this, as mentioned in the previous chapter, graphical symbol can be decomposed by detecting their various meaningful parts called visual primitives, and describe them using graph, where spatial relations can represent edges and shape descriptors can describe nodes. Figure 6.1 provides an idea of how node features can be combined with edge features, where we do not specifically target any problem; it is a generic version of the Attributed Relational Graph (ARG), where both features are combined. For simplicity, in Fig. 6.1, there are just numbers representing signal-based descriptors (quantified) with all required metrical details.

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_6

Fig. 6.1 An example illustrating an attributed relational graph (ARG) via **a** node features; **b** relations and **c** integrating both, for a complete graphical symbol description



(a) Node features

(b) Relational features

(c) Integrating both

6.2 Hybrid Approaches for Graphics Recognition

Let us have a quick review of graphics recognition [1]. This helps to move on to hybrid approaches. In the following, the first part of our discussion will be based on Chap. 4 and second part will be from Chap. 5.

1. A comprehensive review of shape analysis and corresponding tests can be found in [2]. More detailed discussions can be found in Chap. 4. Their context consists of isolated (mainly binary) shapes. In parallel, statistical approaches like global signal-based descriptors [3–8] may not be suitable for distorted images. The primary reason behind it is that small details have been filtered out. As a result, complex and composite graphical symbols cannot be differentiated, where we are required to make a difference between two classes of symbol with a slight change in visual appearance. Not a surprising, the previously mentioned methods may not accommodate with connected or composite symbols (which, in [3]). This happens because of unstable centroid detection and possible occlusions that can lead to build unstable/nonuniform tangents (shape context [6], for instance). In brief, these descriptors may not be appropriate for capturing small details.
   At this point, can we just combine several descriptors or combine classifiers, separately? In the literature, researchers integrated several descriptors [9–11] and combined several classifiers [12] to increase their performance. Such an idea was partially taken from the GREC 1998 [13] that off-the-shelf methods are primarily designed for isolated graphical line symbols. In these statistical approaches, even though statistical signatures are simple to compute and do not suffer from heavy computational cost, their discrimination power and robustness strongly depend on the selection of an optimal set of features. This means their combination does not guarantee expected performance feature selection that varies from one application to another.
2. Besides, another idea is to decompose the symbols into either vector-based primitives, such as points, lines, and arcs or into meaningful parts (regions-of-interest), such as *circles*, *triangles*, and *rectangles*. More detailed explanation can be found in Chap. 5 as such methods fall under structural approaches. With either vector-based primitives or regions-of-interest, one care build attributed relational graphs (ARG) [14, 15], region adjacency graphs (RAG) [16], constraint networks [17], and deformable templates [18]. Their common drawback, however, is related to error-prone raster-to-vector conversion. For example, noisy, deformed, and degraded symbols are affected more. As a consequence, such errors can increase confusions among different classes of symbols. In this context, changes in graph sizes can help fluctuate in computational complexity.
   Several other approaches focused on signature computation by taking meaningful regions-of-interest [19–21]. They speed up matching process in comparison to graph matching. Not a surprising, they are dependent on the region-of-interest detector; we call it visual primitives or meaningful parts/regions.

Other studies integrate spatial relations for symbol recognition: symbol representation and matching techniques of several different structural approaches can be found in [2, 16, 22, 23]. These works are pertinent because structural approaches provide a powerful representation. However, we cannot guarantee that they can provide expected outcomes; and some of their limitations can be addressed integrating with statistical approaches.

On the whole, the conclusion is that one needs an appropriate image description so that the advantages of statistical features can be integrated with the expressiveness of structural approaches. Such a concept can be generic and scalable. Let us have the statement reported by Prof. Tombre, at the event called graphics recognition (GREC) workshop in 2010 [24]:

> *... the very structural and spatial nature of the information we work with makes structural methods quite natural in the community. Their efficient integration into methods which also take full advantage of statistical learning and classification is certainly the right path to take.*

How efficient it could be, if we are able to integrate two different approaches: structural and statistical by just taking their advantages?

An interesting example that uses shape descriptions and relations to form a RAG is found in [25]. In the vector-based RAG description, segmented regions are basically labeled as vertices and geometric properties of adjacency relations are used to label edges. However, we are required to be robust enough so that segmented regions will not be changed even when images are transformed.

In stroke-based hand-drawn symbol recognition, let us consider two previously reported works [26, 27]. One is based on template-based matching and the other one is based on ARG, where the vertices represent geometric primitives, such as lines and arcs, and the edges represent the geometric relationships between the vertices. Matching relies on graph matching or graph isomorphism [28], i.e., conceptually similar to [29]. These methods work well if vertices are well segmented. In their study, online strokes can be easily segmented. Recently, Galois lattice [30] was introduced that aims to classify structural signatures extracted from the Hough transform. These signatures rely on a topological graph, where five topological relations are computed between the segments in addition to their lengths. Note that their length is based on connected and disconnected topological configurations. The first three relations are guided by connected topology, and the remaining two are based on disconnected topology. Their study tells us that Galois lattice-based classification is robust to noise. The method, however, was not tested on the symbols when they are appeared with other graphical elements or with possible texts.

6.3 Integrating Shape with Spatial Relations for Graphics Recognition

Briefly, shape descriptors are appropriate for isolated patterns. In order to work on complex and composite graphical symbols, bag-of-visual words approaches could potentially be used. It, however, needs extensive training sets. In addition, it may

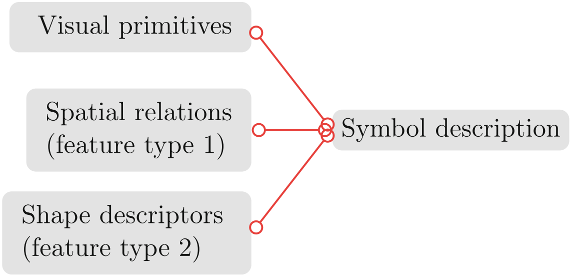

Visual primitives
Spatial relations (feature type 1)
Shape descriptors (feature type 2)
Symbol description

Fig. 6.2 Two different features: **a** spatial relations and **b** shape descriptors are used to describe the graphical symbol. Spatial relations are used to connect (edges in graph) between the visual primitives (nodes) and shape descriptors (signatures) are used to statistically describe what do they look like

not consider/preserve global structure or arrangements of the visual words that are extracted from the studied graphical symbols. Further, human-intuitive visual semantics cannot be seen. Therefore, these approaches are ill-suited in a few cases, such as (a) when large training samples are costlier to collect; (b) when symbols' visual data itself is redundant; and (c) when it is required to check semantics (close-to human description) in symbol description.

In case of real-world data, we are clear that no extensive training set is available (FRESH dataset, i.e., aircraft electric wiring diagrams [31]). More often, only one instance for each symbol class is available. In such a context, the approach can be considered as an appropriate one, if we are able to use a set of well defined, robust, high-level visual part extractors, and segmenting shapes into visual parts, where missing a few does not destroy whole information about the symbol. The driving motivation behind this is that well-mastered, robust, and generic extraction tools can replace statistical bag-of-words learning techniques in case we need more data (for learning).

This results in the extensive use of spatial relations for graphical symbol recognition (Chap. 5: once the symbols are segmented into their meaningful parts/regions, one can use spatial relations between them, where in case of the radial line model (we call RLM) global spatio-structural information can be expressed (but based on visual primitives). Such relations, however, may not explicitly define shape feature in the same way shape descriptors do. The spatial relation descriptors express global pixel distributions between the individual parts (regions-of-interest).

In Sect. 6.4, let us explain how can we construct hybrid approach for graphics recognition (see Fig. 6.2).

6.4 Hybrid Approach on Symbol Description

Let us first describe the visual primitives and their possible changes in shape and size. From their visual appearance, one can understand that the use of shape descriptor (feature) is worth considering in addition to the formation of relative positioning, i.e., spatial relations.

6.4.1 Graph via Visual Primitives

As in Fig. 5.2 of Chap. 5, in the following, we will discuss how can we construct an ARG based on extracted visual primitives (graphical elements) that are connected (via relative spatial relations).

(a) *Visual primitives*:
In general, the visual primitives can be of any kind. The main idea of it is how visually pertinent for the studied application. It can be easily extended or modified by using different vocabularies and other visual cues to adapt to other domains.
Like in earlier chapter, in this study, based on the complexity of the problem (electrical symbols), we have a set of well-controlled visual primitives, and they are *circles*, *corners*, loose end *extremities*, and *thick* (filled). We can express this as a set of visual primitive types: $\sum_{\mathbb{T}} = \{\mathbb{T}_{thick}, \mathbb{T}_{circle}, \mathbb{T}_{corner}, \mathbb{T}_{extremity}\}$. In Fig. 5.2 of Chap. 5, few examples are provided.

(b) *Graph-based representation*:
Instead of taking individual graphical elements to compute possible pairwise spatial relations, it is wise to group them in accordance with the types/classes. In our case, a graphical symbol has four different types of visual primitives, *thick* component, *circle*, *corner*, or *extremity*. Note that the number of elements in one category can be varied from one symbol to another.
To represent a symbol by a complete ARG, let us take a 4-tuple graph: $G = (V, E, F_A, F_E)$, where
V is the set of vertices (primitive types);
$E \subseteq V \times V$ (set of graph edges);
$F_A : V \rightarrow \sum_{\mathbb{T}}$ (a function that assigns attributes to the vertices); and
$F_E : E \rightarrow R_E$ (a function that assigns labels/types to the edges and R refers to spatial relations of the edge E as described in Chap. 5).
Formally, F_A and F_E can be expressed as

$$F_A = \{(\mathbb{T}_1, \mathbb{T}_{thick}), (\mathbb{T}_2, \mathbb{T}_{circle}), (\mathbb{T}_3, \mathbb{T}_{corner}), (\mathbb{T}_4, \mathbb{T}_{extremity})\}, \text{ and}$$
$$F_E = \{((\mathbb{T}_1, \mathbb{T}_2), R_{\mathbb{T}_1,\mathbb{T}_2}), ((\mathbb{T}_1, \mathbb{T}_3), R_{\mathbb{T}_1,\mathbb{T}_3}), ((\mathbb{T}_2, \mathbb{T}_3), R_{\mathbb{T}_2,\mathbb{T}_3})\}, \qquad (6.1)$$

respectively.

For a complete graph, there exist $r = \frac{t(t-1)}{2}$ edges for t attribute types. With this fixed and completely labeled attributes, we are able to avoid the NP-hardness in graph matching [32, 33]. Besides, it also preserves coherence/consistency as visual primitives are semantically different.

(c) *Limitation of ARG and its extension*:
In our set of visual primitives, shape and size of the *thick* components/patterns vary. Variations in their shape and size will not help discriminate their differences by just taking pairwise spatial relations. As a result, the performance may not be optimal [32, 33], which has been clearly mentioned in Chap. 5. In Fig. 6.3, a closer look at the extracted *thick* patterns from different symbols tells us that shape and size of the *thick* pattern vary, and are related to category of the symbol from which they are extracted. For example, a *thick* pattern that is extracted from a junction is visually different from another triangle-shaped ones (diode symbol or from an arrow: ,) in both shape and size. To better distinguish these *think* patterns (in the ARG framework, as mentioned above), shape descriptors can be applied to label vertices:

$$F_A = \{(\mathbb{T}_1, \mathbb{T}_{thick}), \mathscr{S}_{\mathbb{T}_1}), (\mathbb{T}_2, \mathbb{T}_{circle}), (\mathbb{T}_3, \mathbb{T}_{corner}), (\mathbb{T}_4, \mathbb{T}_{extremity})\}. \tag{6.2}$$

In this way, we have observed that vertices are not just labeled by their types but also labeled by statistical signatures (via shape descriptors). However, a single vertex (for one type of visual primitives) via a shape descriptor does not sufficiently exploit shape information, since the elements can be sparse (having different shapes and sizes). Therefore, we focus on *thick* visual primitive type, since the number of *thick* patterns varies a lot from one symbol to another. Under the ARG framework, the vertex labeled with the *thick* primitive type will be split into more specialized *thick* sub-vertices. In general, it can be done separately for all individual *thick* patterns:

$$\mathscr{S}_{\mathbb{T}_1} = \{s_{\mathbb{T}_{1,\kappa}}, \ldots, s_{\mathbb{T}_{1,\mathscr{K}}}\}, \tag{6.3}$$

where $\mathscr{K}$ is the number of *thick* patterns in any studied symbol, which in return results in a graph: $\{G_\kappa\}_{\kappa=1,\ldots,\mathscr{K}}$ ARGs (see Fig. 6.4).

Of course, with description as shown in Fig. 6.4), matching time will basically be increased and it is based on number of *thick* patterns that compose the symbol. To reduce heavy computational time (processing time), let us introduce *thick* pattern clustering.

Clustering of *thick* patterns will definitely reduce the number of graphs since similar *thick* patterns can be averaged to form a single vertex for that particular graph. Not to be confused, clustering refers to the use of shape features so that similar ones can be grouped. Since one does not know how many *thick* patterns are similar in that particular symbol, one has to work on unsupervised clustering.

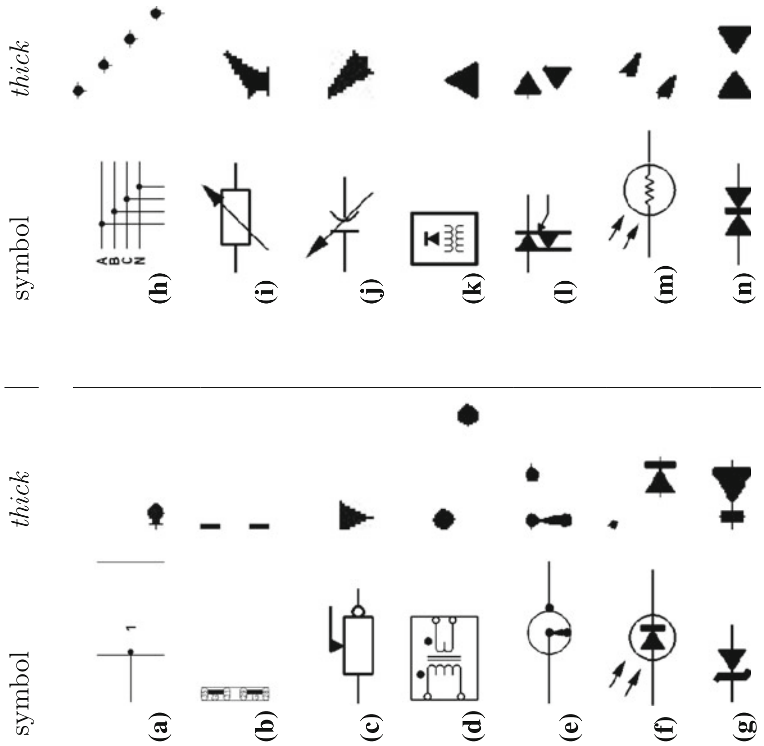


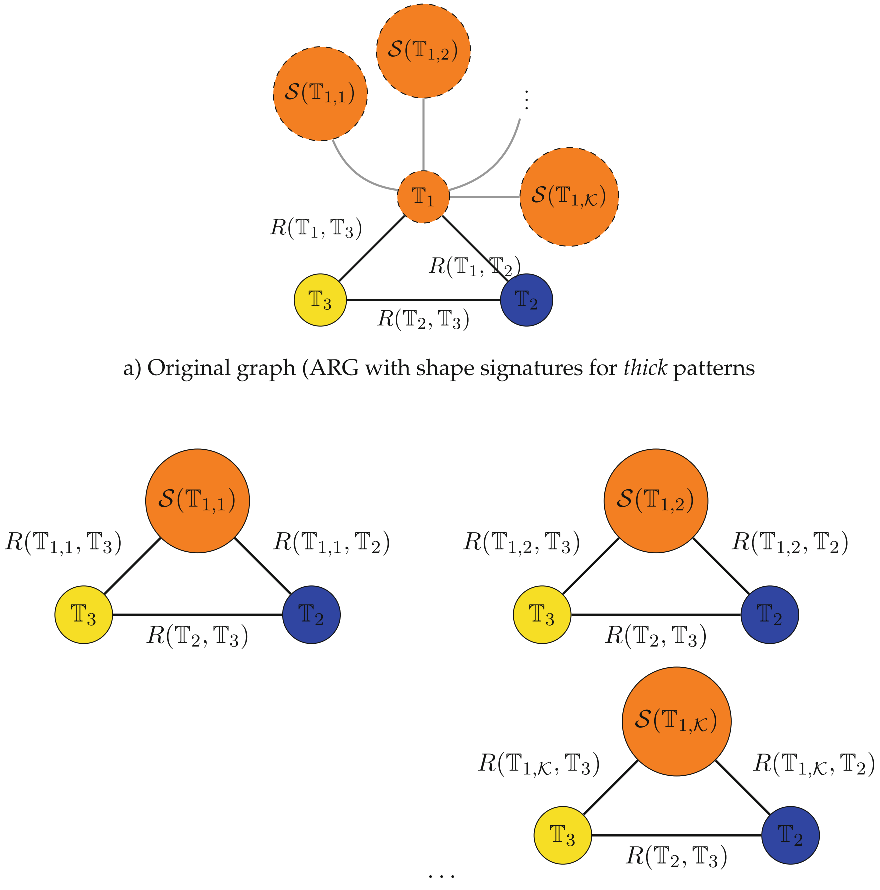
Fig. 6.3 Symbols and their corresponding *thick* patterns, where extracted *thick* patterns are enlarged for better look (visibility)

6.4.2 *Shape-Based Thick Pattern Description in Arg via Clustering*

This primary idea is to work on *thick* pattern clustering based on their appearance. This means that *thick* patterns with different shapes will fall under different clusters or groups with the help of shape descriptors. We have a strong assumption that the *thick* patterns in one cluster are assumed to be extracted from similar types of graphical symbols even though the source can be complex and composite in nature. The discussion is primarily borrowed from the established work reported earlier [1, 34] that is based on PhD report [35, 36].

Let us repeat, since we have no a priori knowledge of the number of shape variations or the number of *thick* patterns in a database, unsupervised clustering is required. In what follows, let us discuss how such a clustering can be handled. In general, two basic steps [37, 38] can be discussed.

(a) *Distance matrix*:
After computing the similarity/dissimilarity scores between all possible pairs of


$\mathcal{S}(\mathbb{T}_{1,2})$
$\mathcal{S}(\mathbb{T}_{1,1})$
$\vdots$
$\mathbb{T}_1$
$\mathcal{S}(\mathbb{T}_{1,\mathcal{K}})$
$R(\mathbb{T}_1, \mathbb{T}_3)$
$R(\mathbb{T}_1, \mathbb{T}_2)$
$\mathbb{T}_3$
$\mathbb{T}_2$
$R(\mathbb{T}_2, \mathbb{T}_3)$
$\mathcal{S}(\mathbb{T}_{1,1})$
$R(\mathbb{T}_{1,1}, \mathbb{T}_3)$
$R(\mathbb{T}_{1,1}, \mathbb{T}_2)$
$\mathbb{T}_3$
$\mathbb{T}_2$
$R(\mathbb{T}_2, \mathbb{T}_3)$
$\mathcal{S}(\mathbb{T}_{1,2})$
$R(\mathbb{T}_{1,2}, \mathbb{T}_3)$
$R(\mathbb{T}_{1,2}, \mathbb{T}_2)$
$\mathbb{T}_3$
$\mathbb{T}_2$
$R(\mathbb{T}_2, \mathbb{T}_3)$
$\mathcal{S}(\mathbb{T}_{1,\mathcal{K}})$
$R(\mathbb{T}_{1,\mathcal{K}}, \mathbb{T}_3)$
$R(\mathbb{T}_{1,\mathcal{K}}, \mathbb{T}_2)$
$\mathbb{T}_3$
$\mathbb{T}_2$
$R(\mathbb{T}_2, \mathbb{T}_3)$
...

a) Original graph (ARG with shape signatures for *thick* patterns

b) Graph decomposition (based on number of *thick* patterns

Fig. 6.4 ARG description from three visual primitives (*thick*, *corner* and *extremity*, for example): **a** the original graph is split into **b** several graphs (based on number of *thick* patterns)

thick patterns, a distance (similarity or dissimilarity) matrix can be constructed, which we often call: distance matrix.

Let us represent a *thick* pattern p by a signature (shape) $\mathtt{s}_p$ of size i. For *thick* patterns: a and b, $\delta(\mathtt{s}_a, \mathtt{s}_b)$, the distance (similarity or dissimilarity) metric between the signatures can be computed. Beside the use of shape descriptor, the outcome relies on the metric used. A few but, obvious metrics are

$$\delta(\mathtt{s}_a, \mathtt{s}_b) = \begin{cases} \sum_i |\mathtt{s}_a[i] - \mathtt{s}_b[i]| \text{ (City-block),} \\ \sqrt{\sum_i (\mathtt{s}_a[i] - \mathtt{s}_b[i])^2} \text{ (Euclidean), and} \\ \sum_i (\mathtt{s}_a[i] - \mathtt{s}_b[i])^2 \text{ (Squared euclidean).} \end{cases} \tag{6.4}$$

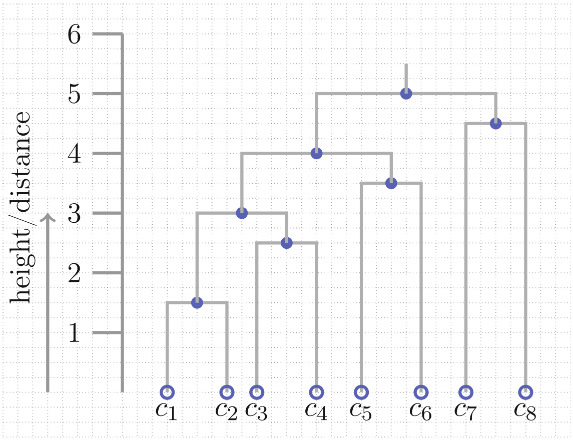


Fig. 6.5 Dendrogram example using eight *thick* patterns (labeled with clusters $c_1, c_2, \ldots, c_8$). They merged based on their similarity (see height/distance between them to know which are merged first)

There are different ways to select "which combination" of shape descriptor and metric yields the better results, and it is basically based on the following two issues.

(b) *Merge/cluster patterns*:
Similar *thick* patterns can be clustered/grouped that can be in the form of a hierarchical cluster tree. Other methods/techniques can be employed.
Considering similarity matrix (via computed distance between shape descriptors), for our problem, we can work on measuring their distances, i.e., how far one pattern is from others? This means that *linkage methods* could potentially be a better choice for our discussion. Note that in the literature, broadly speaking, we have three different types of linkage methods, and all of them rely on the distance matrix:

(i) Single-linkage clustering (SLC),
(ii) Complete-linkage clustering (CLC), and
(iii) Average-linkage clustering (ALC).

SLC is also known by *nearest neighbor* clustering. This means that the most closest ones are merged to form a cluster. CLC takes the maximum distance between the two clusters. ALC uses the mean distance between elements of each cluster. Mathematically, the distance between two clusters c_a and c_b can be expressed as

$$D(c_a, c_b) = \begin{cases} \min\{\,\delta(\mathrm{s}_a, \mathrm{s}_b) : \mathrm{s}_a \in c_a,\ \mathrm{s}_b \in c_b\} \text{ (SLC)}, \\ \max\{\,\delta(\mathrm{s}_a, \mathrm{s}_b) : \mathrm{s}_a \in c_a,\ \mathrm{s}_b \in c_b\} \text{ (CLC), and} \\ \dfrac{1}{|c_a| \times |c_b|} \displaystyle\sum_{\mathrm{s}_a \in c_a} \sum_{\mathrm{s}_b \in c_b} \delta(\mathrm{s}_a, \mathrm{s}_b) \text{ (ALC).} \end{cases} \tag{6.5}$$

In our specific context, we consider an agglomerative hierarchical clustering scheme, where the similarity matrix will be used. Technically, it deals with erasing rows and columns in this similarity matrix each time clusters are grouped. Grouping/clustering happens based on the similarity score, where the selected metric and linkage method may change the outcome. If two patterns are merged, similarity matrix will be update by erasing the rows and columns. This will also replace the merged distance values by the linkage values. Until all clusters are merged or it reaches a preset cluster threshold, the process repeats.

Figure 6.5 shows an example of a dendrogram using agglomerative hierarchical clustering. In Fig. 6.5, we observe that a single cluster is the outcome of the whole clustering process. Not to be confused, having a single cluster at the output is not our aim. The similarity between pairs is simply taken from the linkage distance computation technique that can be either SLC, CLC, or ALC. For instance, clusters c_1 and c_2 are merged at a distance of 1.5. This is also called the *dendrogrammatic distance*.

The whole clustering process does not answer two questions: cluster verification and validation. These are essentials because of the following reasons:

(a) It will check all possible pairs between both descriptor metrics and linkage methods, and it will suffer from high time complexity issue. To avoid this, we need to check whether we are able to compute an optimal combination via *cluster verification.*

(b) How can we find the optimal number of clusters? Can we set cut-off threshold to stop the process of agglomerative hierarchical clustering?
In Fig.??, there exists no cut-off threshold; instead, it provides a whole clustering process. Cut-off threshold refers to the appropriate number of clusters at the output. In this case, one of the two different conditions can be considered: (a) a manual threshold and (b) an automatic threshold via cluster validation techniques. The latter condition is interesting, since manual threshold cannot be generic. For cluster validation, either unsupervised or supervised approaches can be employed. In our context, *unsupervised cluster validation* is appropriate.

6.4.3 Cluster Verification and Validation

(a) *Cluster verification*:
Cluster analysis depends on

   (a) shape descriptors,
   (b) distance metric, and
   (c) linkage function.

This means that different results can happen from different pairs of distance metric and linkage technique. Besides, since the optimal combination depends on what shape signatures/descriptors and data are being considered, cluster verification is required. Therefore, an obvious solution is that the *cophenetic correlation coefficient* [39–41] can be used. This lets us choose the best pair.

In hierarchical clustering, the height of the link is known by cophenetic distance that represents the distance between two clusters. For any original data $\mathscr{S} = \{\mathtt{s}_i\}$, we produce dendrogram Z after the clustering process. Suppose $\bar{\delta}$ is the average value of all distance measures $\delta\left(\mathtt{s}_i, \mathtt{s}_j\right)$ between the data samples and $\bar{z}$ be the average of the $Z_{i,j}$ (the dendrogrammatic distance between the data samples). The cophenetic correlation coefficient (CCC) then can be computed as [41]

$$\mathrm{CCC} = \frac{\sum\limits_{i<j} \left(\delta\left(\mathtt{s}_i, \mathtt{s}_j\right) - \bar{\delta}\right)\left(Z_{i,j} - \bar{z}\right)}{\sqrt{\left[\sum\limits_{i<j} \left(\delta\left(\mathtt{s}_i, \mathtt{s}_j\right) - \bar{\delta}\right)^2\right]\left[\sum\limits_{i<j} \left(Z_{i,j} - \bar{z}\right)^2\right]}}. \tag{6.6}$$

The dendrogrammatic distance is the height of the node at which these two points are first merged (see Fig. 6.5 for better understanding).

Note that we have agreed on use of the CCC. It tells us a combined measure between two different sets of values: (i) distance metric and (ii) linkage function. In other words, if the clustering is valid, the linking of patterns in a cluster tree produces a strong correlation with the distance between the clusters. Ideally, for the accurately clustered patterns, the CCC value is equal to 1. Considering a set of arbitrary features, we are able to illustrate (i) how the cluster verification works and (ii) how we can we achieve the best combination of a distance metric and a linkage method (see Table 6.1, as an example). In Table 6.1, CCC values are provided for all possible pairs. In this example, like we have mentioned earlier, the Euclidean distance metric and the average-linkage clustering technique is found to be the best compared to others, since the CCC value is close to 1. This helps us that no remaining pairs need to process for cluster validation. Unlike in our illustration (see Table 6.1), the number of pairs can be increased if we employ more distance metrics.

(b) *Cluster validation*:

Determining the expected number of clusters is our concern, since number of clusters has an effect on the recognition performance. For example,

- If many clusters are at the output, cluster size will be small and their elements will be highly similar and consistent. This also means their inter-cluster elements are close to each other. Further, the more the number of clusters, the more the time complexity. It is also sensitivity to noise.
- In an opposite case, if fewer clusters are at the output, they will then automatically be larger in size. This means that intra-cluster distance can be large

Table 6.1 CCC from all possible pairs of distance metric and clustering linkage methods (see data in Fig. 6.6). The best pair is the one from which the CCC produces value that is closer to 1

Techniques	City-block	Euclidean	Squared Euclidean
SLC	0.8469	0.8738	0.8203
CLC	0.8460	0.8720	0.8203
ALC	0.8560	**0.8833**	0.8240

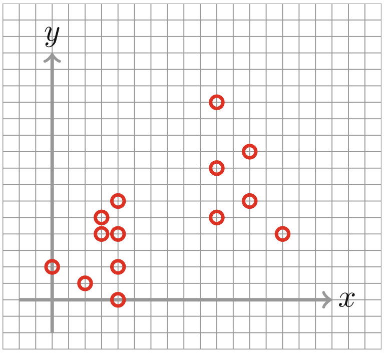


Fig. 6.6 An example of a data: a set of 2D points. A few 2D points are considered to simplify the problem

i.e., their elements do not have similar appearance. They, however, are more robust to noise since they do not take detailed information about the shape.

We have mentioned earlier that the evaluation measures that are applied to check several aspects of cluster validity are traditionally grouped into two approaches: supervised and unsupervised.

Supervised measures often take external indices, since they use additional information. In our problem, since we do not have external input to determine the number of clusters, unsupervised technique is only way to proceed. Unsupervised measures of cluster validity are often based on (i) cluster cohesion and (ii) separation that are under the framework of internal indices.

(i) Cluster cohesion refers to how tight is the cluster (compactness). In other words, it expresses how closely related the objects in a cluster are.
(ii) Cluster separation refers to how distinct or well separated a cluster is from other clusters. It is assumed that the clusters themselves should be widely separated. Three common approaches that measure the distance between two different clusters are (i) the closest member (in the clusters), (ii) the most distant members, and (iii) the centers of the clusters.

Within the framework, we have several different indices to validate the clusters in the literature. In this chapter, in order just to understand the concept, let us use a few of them but well-known indices:

(a) Dunn index,
(b) Davies–Bouldin index,
(c) Silhouette index, and
(d) Score function.

Dunn Index (DU):

It is the ratio between the minimal intra-cluster distance to maximal inter-cluster distance [42]. The DU for k clusters can be computed as

$$\begin{aligned} \mathrm{DU}_k &= \min_{i=1,\ldots,k}\left\{\min_{j=i+1,\ldots,k}\left(\frac{\mathrm{dist.}(c_i, c_j)}{\max\limits_{m=1,\ldots,k} \mathrm{dist.}(c_m)}\right)\right\}, \quad (6.7)\\ \text{and } \mathrm{dist.}(c_i, c_j) &= \min_{s_a \in c_i, s_b \in c_j} \delta(s_a, s_b) \text{ and}\\ \mathrm{dist.}(c_m) &= \max_{s_a, s_b \in c} \delta(s_a, s_b). \end{aligned}$$

As expected, for large inter-cluster distance and small intra-cluster distance, it produces the maximum value.

Davies–Bouldin Index (DB):

It determines both: inter-class distance and their individual compactness [43]. The DB index can be computed as

$$\begin{aligned} \mathrm{DB}_k &= \frac{1}{k}\sum_{i=1}^{k} \max_{j=1,\ldots,k, i\neq j}\left\{\frac{\mathrm{dist.}(c_i) + \mathrm{dist.}(c_j)}{\delta(c_i, c_j)}\right\}, \quad (6.8)\\ \text{and } \mathrm{dist.}(c_i) &= \frac{1}{n_i}\sum_{s_a \in c_i} \delta(s_a, s_i^{mean}), \end{aligned}$$

where n_i is the number of elements and s_i^{mean} is the centroid of cluster c_i. The DB index is expected to be small for the best number of clusters.

Silhouette Index (SI):

It computes the following: (i) silhouette width for each sample, (ii) average silhouette width for each cluster, and (iii) overall average silhouette width for a total dataset [44]. The silhouette takes cluster tightness and separation into account. The average silhouette width helps decide how good the clusters are. Since it is an average of all observations, it can be computed as

$$\mathrm{SI}_k = \frac{1}{n}\sum_{i=1}^{n} \frac{(\mathrm{dist}\widehat{2}._i - \mathrm{dist}\widehat{1}._i)}{\max(\mathrm{dist}\widehat{1}._i, \mathrm{dist}\widehat{2}._i)}, \quad (6.9)$$

where n is the total number of elements, $\mathrm{dist}\widehat{1}._i$ is the average distance between the element i and all other elements in its own cluster, and $\mathrm{dist}\widehat{2}._i$ is the minimum of the average distance between i and elements in other clusters. It maximizes its value for the best output.

Score Function (SF):

Like the two other indices: DI and DB, SF [45] also relates inter-class and intra-class distances, and can be formalized as

(i) Between class distance (*bcd*), and
(ii) Within class distance (*wcd*).

Inter-class distance, i.e., bcd, can mathematically be computed as follows:

$$bcd = \frac{\sum_{i=1}^{k} \delta(s_i^{mean}, s_{tot.}^{mean}) \times n_i}{n \times k}, \tag{6.10}$$

where k is the number of clusters of size n, s_i^{mean} is the centroid of cluster c_i having n_i elements, and $s_{tot.}^{mean}$ is the centroid of all clusters.

In a similar fashion, intra-class distance, i.e., wcd can mathematically be computed as

$$wcd = \sum_{i=1}^{k} \left(\frac{1}{n_i} \sum_{s_a \in c_i} \delta(s_a, s_i^{mean}) \right). \tag{6.11}$$

By taking bcd and wcd into account, SF can be computed as

$$\mathrm{SF} = 1 - \left(1/e^{e^{bcd-wcd}}\right). \tag{6.12}$$

The higher the value of the SF, the more appropriate the number of clusters. In other words, it maximizes the *bcd* and minimizes the *wcd*, which is expected.

To understand how cluster validation indices work, we take an example from Fig. 6.7, where best number of clusters are presented. In Fig. 6.7, we observe that all cluster validation indices produce two clusters at the output.

6.5 Experiments

6.5.1 Graphical Symbol Recognition

The recognition framework principally follows the corresponding relation matching (presented earlier, see Chap. 5) for matching two graphs: $G1$ and $G2$, where $G(V, E, F_A, F_E)$. Then, their distance is computed as

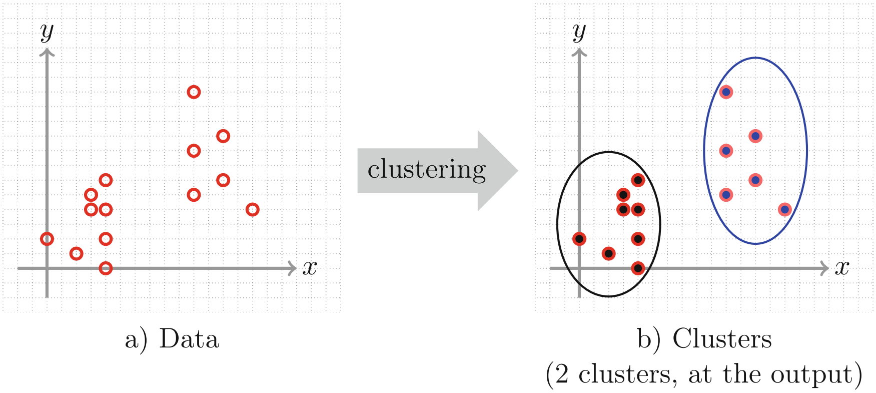


Fig. 6.7 Cluster validation: like it is mentioned in Table 6.2, two clusters are at the output (from all indices), where it started with six clusters in the beginning

Table 6.2 Cluster validation: two clusters are at the output (from all indices), where it started with six clusters in the beginning

Indices	Clusters					
	1	2	3	4	5	6
Silhouette index	—	**0.7650**	0.6523	0.6689	0.7059	0.7983
Dunn index	—	**1.2878**	0.6695	0.7906	0.7071	0.5590
Davies–Bouldin index	—	**0.0359**	0.0984	0.1329	0.0725	0.0428
Score function index	0.0000	**1.0000**	0.9000	0.8700	0.5620	0.5200

$$\text{Dist.}(G1, G2) = \sum_{r \in E} \delta\left(F1_E(r), F2_E(\sigma(r))\right), \tag{6.13}$$

where

- $\delta(,)$ = distance between two relational signatures,
- $F1$ (respectively $F2$) = function that computes relational signature of an edge, and
- $\sigma : E1 \rightarrow E2$ = function that maps edges from one graph to the other.

In this chapter, graphical symbol description has been enriched by adding signal-based (shape) node features. More specifically, a symbol $\mathscr{S}$ has $\mathscr{K}$ number of *thick*s and we have a set of $\{G_\kappa\}_{\kappa=1}^{\mathscr{K}}$ ARGs representing it and $\mathscr{K}$ varies from one symbol to another.

To compute the similarity between the two symbols, a query symbol $\mathscr{S}^q = \{G_\kappa^q\}_{\kappa=1}^{\mathscr{K}}$ and a database symbol $\mathscr{S}^{db} = \{G_{\kappa'}^{db}\}_{\kappa'=1}^{\mathscr{K}'}$, the main idea is to find the best matched graphs pair, i.e., distance between them is small (or close to zero). To find similarity between the symbols, $\mathscr{S}^q$ and $\mathscr{S}^{db}$, their corresponding graphs, G_κ^{db} (database) G_κ^q (query), are matched. For a pair of symbols $\mathscr{S}^\dagger$ and $\mathscr{S}^\ddagger$, we can formally compute the minimum distance as follows:

$$\Delta(\mathscr{S}^\dagger, \mathscr{S}^\ddagger) = \min_\kappa \left(\min_{\kappa'} \left(\text{Dist.}(G_\kappa^\dagger, G_{\kappa'}^\ddagger) \right) \right). \tag{6.14}$$

In Fig. 6.8, we have three possible cases to realize graph matching, where *thick* pattern clustering is not considered. The closest candidate for any query symbol $\mathscr{S}^q$ in whole database symbols $\{\mathscr{S}^{db}\}_{db=1,\ldots,\mathbb{DB}}$ can be computed as $\min_{db} \left(\Delta\left(\mathscr{S}^q, \mathscr{S}^{db}\right)\right)$. We are not just limited to select the closest candidate (recognition); retrieval is possible, where database symbols are ranked based on the similarity score.

Like other graph matching procedure (mentioned in previous chapters), it may suffer from heavy computational complexity. But an inclusion of *thick* pattern clustering can help reduce time complexity in graph matching.

For each query symbol, the first step is to select the cluster in which the query thick belongs to. For distance computation, between any test *thick* pattern, centroid (signature) of the particular cluster, i.e., $\delta(s_a, s_i^{mean})$, is used. With this process, more than one cluster can be selected if a query symbol has multiple *thick* patterns with a different appearance (shape). Once we select cluster(s), the symbols related to those *thick* patterns (corresponding symbols) are taken for graph matching. Note that graph matching is explained in Eq. (6.13).

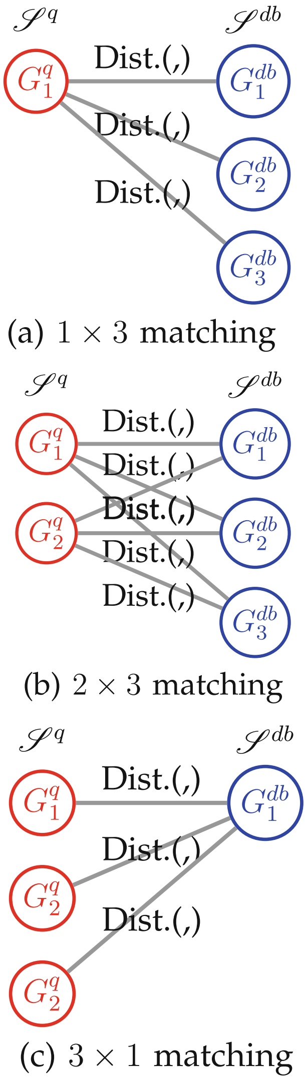
6.5.2 Results

For shape-based signatures (to label nodes), let us employ exactly similar set of shape descriptors as reported in Chap. 4 in this *thick* pattern clustering mechanism: (a) Zernike moments (ZM) [4], (b) $\mathscr{R}$-signature [5], (c) Shape context (SC) [6], (d) Generic Fourier descriptor (GFD) [7], and (e) DTW-Radon [46, 47].

With the use of clustering of *thick* patterns, we aim to take similar ones in a cluster that eventually increase retrieval performance via corresponding relational signature matching. Let us remind that radial line model is used to produce signature, as discussed in Chap. 5. This means that the current approach is the combined version of both chapters: Chaps. 4 and 5.

Not to be away from the real-world problem and to make a fair comparison with the results (before), complex and composite graphical symbol dataset (see Fig. 4.3 of Chap.4) is considered for the test. This means that the evaluation metric, we call retrieval efficiency [32, 33, 48, 49], is used as a measure of retrieval quality. In such a context, let us broadly include two major objectives:

(a) Check how shape descriptors and cluster validation indices affect on the clusters at the output, and



(a) 1×3 matching

(b) 2×3 matching

(c) 3×1 matching

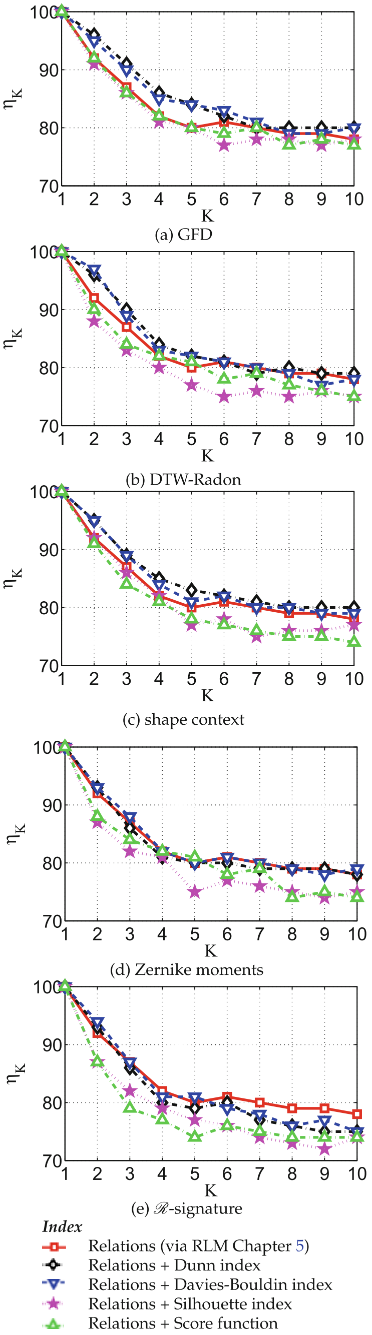
Fig. 6.8 Between two symbols: $\mathscr{S}^q$ and $\mathscr{S}^{db}$, few graph matching schemes are shown. Each encircled token represents a graph. The basic idea is to illustrate how simple it is, without integrating *thick* pattern clustering

(b) Check how *thick* pattern clustering can help enrich the symbol description (through experimental results).

Following the reported results in Fig. 5.18 of Chap. 5, it is a time to check whether inclusion of *thick* pattern selection (plus relations) improves the performance of the system. Since clustering performance relies on signatures (shape) and cluster validation indices, it is wise to consider into account.

Figure 6.9 shows the comparison of performance of cluster validation indices for different shape descriptors. In these tests, retrieval performances on a one-to-one

Fig. 6.9 Average retrieval efficiency (requested list: 1−10) using signal-based shape descriptors (aimed for *thick* patterns clustering) and several different cluster validation indices



(a) GFD

(b) DTW-Radon

(c) shape context

(d) Zernike moments

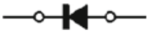
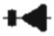
(e) $\mathscr{R}$-signature

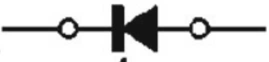
basis can be observed and analyzed. In brief, GFD outperforms all, but DTW-Radon performs almost equally having a marginal difference. Zernike moments, shape context, and $\mathscr{R}$-signature are lagging behind. It is important to note that these results are based on the dataset (particular to electrical circuit diagram) and are completely based on how *thick* patterns are extracted. Therefore, one cannot draw conclusion on which descriptor performs the best, as their performances vary with the changes in datasets, i.e., studied samples.

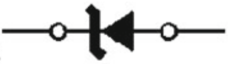
Shape descriptor selection does not end the process. It also needs to account cluster validation indices since we know that results may be changed in accordance with the change in cluster validation index. As shown in Fig. 6.9, for all shape descriptors, two indices, Dunn and Davies–Bouldin, provide almost similar advancements, while the remaining indices do not. Therefore, for this dataset, either Dunn or Davies–Bouldin index can be considered.

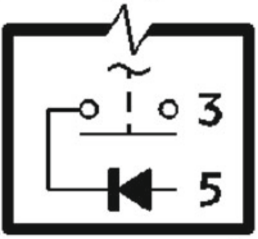
Considering such a dataset, we observe that *thick* pattern selection via clustering advances retrieval performance in addition to relational signature matching. However, the difference is marginal. This means that it does not always guarantee the increment in performance, since difference is not statistically significant. The primary reason behind this is that not all query symbols contain *thick* pattern in their visual primitive sets. In other words, the absence of *thick* visual primitive type means ranking has been made only through relational signature matching. For a quick visual illustration and to check where it has been advanced, let us provide a qualitative example.

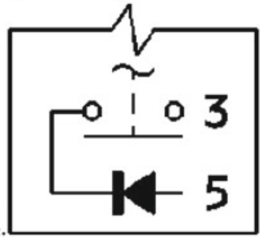
Example

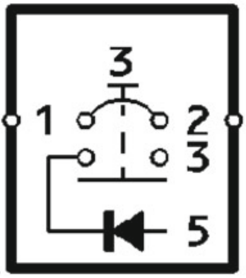
For a query  , DTW-Radon (for labeling nodes, *thick*),  , the first five ranked database symbols retrieved are as follows:

1. 

2. 

3. 

4. 

5. 

6.6 Conclusions

In this chapter, we have discussed a detailed study of several different (but, major) hybrid approaches that are designed for graphics recognition, i.e., graphical symbol recognition, retrieval, and spotting. More specifically, we have explained a complete concept of how the shape signatures of the extracted visual primitives (showing significant shape variations) can be integrated with spatial relations that exist between them. Inspired from the GREC'10 [24] and real-world problem, the concept has been tested to see how well it works. Further, we have discussed on unsupervised clustering, where a very specific visual primitive i.e., *thick* pattern is taken into account. Note that the clustering of *thick* patterns thus opens a generic idea that it can be applied for other visual primitives (and even for other image recognition problems). In short, this chapter aims to bring an attention to the use of a hybrid approach in graphics recognition since it combines both worlds: structural and statistical; and more importantly, they complement each other.

In the next chapter, let us discuss syntactic approaches for graphics recognition.

References

1. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
2. L.P. Cordella, M. Vento, Symbol and shape recognition, in *Graphics Recognition, Recent Advances*, ed. by A.K. Chhabra, D. Dori, Lecture Notes, in Computer Science, vol. 1941, (Springer, Berlin, 2000), pp. 167–182
3. G.-C. Feng, Y.Y. Tang, Pong Chi Yuen, Printed chinese character similarity measurement using ring projection and distance transform. Int. J. Pattern Recognit. Artif. Intell. **12**(2), 209–221 (1998)
4. W.-Y. Kim, Y.-S. Kim, A region-based shape descriptor using zernike moments. Signal Process. Image Commun. **16**(1–2), 95–102 (2000)
5. S. Tabbone, L. Wendling, J.-P. Salmon, A new shape descriptor defined on the radon transform. Comput. Vis. Image Underst. **102**(1), 42–51 (2006)
6. S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Anal. Mach. Intell. **24**(4), 509–522 (2002)
7. D. Zhang, L. Guojun, Shape-based image retrieval using generic fourier descriptor. Signal Process. Image Commun. **17**(10), 825–848 (2002)
8. D. Zhang, G. Lu, Review of shape representation and description techniques. Pattern Recognit. **37**(1), 1–19 (2004)
9. J.P. Salmon, L. Wendling, S. Tabbone, Improving the recognition by integrating the combination of descriptors. Int. J. Doc. Anal. Recognit. **9**(1), 3–12 (2007)
10. O.R. Terrades, E. Valveny, S. Tabbone, On the combination of ridgelets descriptors for symbol recognition, in *Graphics Recognition. Recent Advances and New Opportunities*, ed. by L. Wenyin, J. Lladós, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 5046, (Springer, Berlin, 2008), pp. 40–50
11. S. Barrat, S. Tabbone, A bayesian network for combining descriptors: application to symbol recognition. Int. J. Doc. Anal. Recognit. **13**(1), 65–75 (2010)
12. E. Valveny, S. Tabbone, Oriol Ramos Terrades, Optimal classifier fusion in a non-bayesian probabilistic framework. IEEE Trans. Pattern Anal. Mach. Intell. **31**(9), 1630–1644 (2009)

13. K. Tombre, C. Ah-Soon, P. Dosch, A. Habed, G. Masini, Stable, robust and off-the-shelf methods for graphics recognition. Proc. IAPR Int. Conf. Pattern Recognit. **1**, 406 (1998)
14. H. Bunke, B.T. Messmer, *Efficient attributed graph matching and its application to image analysis, in CIAP* (Springer, London, 1995), pp. 45–55
15. D. Conte, P. Foggia, C. Sansone, M. Vento, Thirty years of graph matching in pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **18**(3), 265–298 (2004)
16. J. Lladós, E. Martí, J.J. Villanueva, Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. IEEE Trans. Pattern Anal. Mach. Intell. **23**(10), 1137–1143 (2001)
17. C. Ah-Soon, K. Tombre, Architectural symbol recognition using a network of constraints. Pattern Recognit. Lett. **22**(2), 231–248 (2001)
18. E. Valveny, E. Martí, A model for image generation and symbol recognition through the deformation of lineal shapes. Pattern Recognit. Lett. **24**(15), 2857–2867 (2003)
19. P. Dosch, J. Lladós, *Vectorial signatures for symbol discrimination, in Graphics Recognition*, Lecture Notes in Computer Science Series (Springer, Berlin, 2003), pp. 154–165
20. L. Wenyin, W. Zhang, L. Yan, An interactive example-driven approach to graphics recognition in engineering drawings. Int. J. Doc. Anal. Recognit. **9**(1), 13–29 (2007)
21. M. Rusiñol, A. Borràs, J. Lladós, Relational indexing of vectorial primitives for symbol spotting in line-drawing images. Pattern Recognit. Lett. **31**(3), 188–201 (2010)
22. L.P. Cordella, M. Vento, Symbol recognition in documents: a collection of techniques? Int. J. Doc. Anal. Recognit. **3**(2), 73–88 (2000)
23. J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition: current advances and perspectives, in *GREC - Algorithms and Applications*, ed. by D. Blostein, Y.-B. Kwon, Lecture Notes, in Computer Science, vol. 2390, (Springer, Berlin, 2002), pp. 104–127
24. K. Tombre, Graphics recognition - what else?, in *Graphics Recognition. Achievements, Challenges, and Evolution*, ed. by J.-M. Ogier, W. Liu, J. Lladós, Lecture Notes, in Computer Science, vol. 6020, (Springer, Berlin, 2010), pp. 272–277
25. P. Le Bodic, H. Locteau, S. Adam, P. Héroux, Y. Lecourtier, A. Knippel, Symbol detection using region adjacency graphs and integer linear programming, in *Proceedings of International Conference on Document Analysis and Recognition* (2009), pp. 1320–1324
26. L.B. Kara, T.F. Stahovich, An image-based, trainable symbol recognizer for hand-drawn sketches. Comput. Graph. **29**(4), 501–517 (2005)
27. W.S. Lee, L.B. Kara, T.F. Stahovich, An efficient graph-based recognizer for hand-drawn symbols. Comput. Graph. **31**(4), 554–567 (2007)
28. B.T. Messmer, H. Bunke, Efficient subgraph isomorphism detection: a decomposition approach. IEEE Trans. Knowl. Data Eng. **12**(2), 307–323 (2000)
29. X. Xiaogang, S. Zhengxing, P. Binbin, J. Xiangyu, L. Wenyin, An online composite graphics recognition approach based on matching of spatial relation graphs. Int. J. Doc. Anal. Recognit. **7**(1), 44–55 (2004)
30. M. Coustaty, K. Bertet, M. Visani, J.-M. Ogier, A new adaptive structural signature for symbol recognition by using a galois lattice as a classifier. IEEE Trans. Syst. Man Cybern.- Part B Cybern. **41**(4), 1136–1148 (2011)
31. M. Tooley, D. Wyatt, *Aircraft Electrical and Electronic Systems: Principles, Operation and Maintenance, Aircraft engineering principles and practice* (Butterworth-Heinemann, Amsterdam, 2008)
32. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
33. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. Artif. Intell. **28**(06), 1450017 (2014)
34. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2011), pp. 228–237
35. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis). Ph.D. thesis, University of Lorraine, France (2011)

36. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, New York, 2015), pp. 1–22
37. A.K. Jain, R.C. Dubes, *Algorithms for Clustering Data* (Prentice-Hall Inc., Upper Saddle River, 1988)
38. A.P. Reynolds, G. Richards, B. Iglesia, V.J. Rayward-Smith, Clustering rules: a comparison of partitioning and hierarchical clustering algorithms. J. Math. Model. Algorithms **5**, 475–504 (2006)
39. R.R. Sokal, F.J. Rohlf, The comparison of dendrograms by objective methods. Taxon **11**(2), 33–40 (1962)
40. F.J. Rohlf, D.L. Fisher, Test for hierarchical structure in random data sets. Syst. Zool. **17**, 407–412 (1968)
41. D.B. Carr, C.J. Young, R.C. Aster, X. Zhang, *Cluster analysis for ctbt seismic event monitoring* (In a study prepared for the U.S, Department of Energy, 1999)
42. J.C. Dunn, Well-separated clusters and optimal fuzzy partitions. J. Cybern. **4**(1), 95–104 (1974)
43. D.L. Davies, D.W. Bouldin, A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. PAMI **1**(2), 224–227 (1979)
44. P. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. J. Comput. Appl. Math. **20**(1), 53–65 (1987)
45. S. Saitta, B. Raphael, I.F. Smith, A bounded index for cluster validity, in *Proceedings of International Conference on Machine Learning and Data Mining in Pattern Recognition* (Springer, Berlin, 2007), pp. 174–187
46. K.C. Santosh, B. Lamiroy, L. Wendling, DTW for matching radon features: a pattern recognition and retrieval method, in *Advances Concepts for Intelligent Vision Systems (ACIVS)* (2011), pp. 249–260
47. K.C. Santosh, B. Lamiroy, L. Wendling, Dtw-radon-based shape descriptor for pattern recognition. Int. J. Pattern Recognit. Artif. Intell. **27**(3), 1350008 (2013)
48. M.S. Kankanhalli, B.M. Mehtre, W. Jian, Kang, Cluster-based color matching for image retrieval. Pattern Recognit. **29**, 701–708 (1995)
49. K.C. Santosh, Complex and composite graphical symbol recognition and retrieval: a quick review, in *Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers*, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi (Science, Communications in Computer and Information, 2017), pp. 3–15

Chapter 7
Syntactic Approaches

7.1 Syntactic Approaches-Based Graphical Symbol Recognition

This section provides a quick review of syntactic approaches-based graphical symbol recognition. Considering theories and models, in semantic approaches, syntactic expressions are found to be powerful descriptions. Like semantic descriptions, these are language independent. Interestingly, for syntactic descriptions, nonstandard models do not pose any issue. In other words, syntactic descriptions can be made as simple as semantic ones. Further, scientific theories can be drawn by syntactic descriptions [1].

In graphics recognition [2], the syntactic approaches rely on grammar-based (or rule-based) formal notions of composition. For example, graph grammars have been widely adopted concept in describing 2D graphical symbol or pattern [3, 4]. Even though we consider them as powerful tools, they require extensive preprocessing in case of noisy data, where the rule-set (grammar) has to accurately describe the shape of the pattern/symbol. For a quick reference about array, tree, and graph grammars, let us refer to the previous work by Rosenfeld [5]. Graph grammars are first used to perform diagram recognition. They were limited to local 2D patterns and/or structures. Interestingly, graph grammars are not limited to 2D and/or isolated patterns; it can also be extended to interpret electrical circuit diagrams, i.e., complex and composite graphical symbols. For the same purpose, the build-weed-incorporate (BWI) programming was introduced later. BWI provides a concept about interaction or possible relations among distant symbols (physically) and such a property is semantically crucial. To know more about graph grammars and their genericity, let us refer to previous work [6].

Like it is mentioned earlier, human design with standard rules, i.e., semantics, can help build computer-aided design (CAD) projects. The quotation of industrial designs, for example, follows specific rules. Rules follow either the 2D grammars [7, 8] or the plex grammars [9]. Labeled graph (but, undirected) representation—which we call web—was explained well in [7] to syntactically analyze the dimensions.

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_7

Using the conventions of web grammar, it is possible to construct a set of rules. This means that they specify how well all dimension sets can be produced. Plex grammars are powerful and can be generalized because they are able to combine the strings-, trees-, and web-grammars as sub-cases. Integrating relations that exist between visual primitives or cues in a string in a grammar are crucial. But concatenation does not always produce expected results and the effective way to integrate/combine relations into string can be more standard. The generic schema, however, can be restricted by the complexity of the problem [10].

Using grammars, there has always been an attraction behind the projects, such as diagram interpretation and full syntactic analysis of engineering drawings [11, 12]. Often, attributed or labeled graph grammars and possible applications to the schematic diagrams interpretation were introduced in [11], where the grammar was used to extract a description. Not limited to schematic diagrams, two other classes of diagrams were studied: flowcharts and circuit diagrams. In [12], engineering drawing interpretation was introduced. It is based on the combination of schemata expressing their prototypes, where construction was based on a library of image analysis (fairly low-level) and a set of control rules with the use of parser.

Graph grammars can be supported by the use of textures in architectural plans, where we observed that texture analysis and recognition is important [13]. Models that are based on the texture features can add another level of image understanding. The parser then helps decide whether the grammar accepts the symbol description by the use of the repetitive structured patterns with different textures. A region adjacency graph (RAG) representation can be transformed into the grammar for all vectorized documents. After that, in [14], within the framework, as an extension of the previous work, the clustering that is based on representatives through shape analysis. Few other approaches were used in the process of vectorization [15] so that data simplification by taking basic shapes is possible. This makes subsequent processes easy. Integrating structural and syntactic approaches is always an attraction that advances the recognition of graphical elements or symbols in the technical documents [16].

Considering the vector-based symbol recognition, syntactic symbol recognition concept was introduced by Yajie et al. 2007 [17]. In contrast to state-of-the-art approaches, where geometric relations among primitives are used, their method employs a model that can describe geometric information of all visual primitives or cues. This, of course, relates to the whole symbol description that is based on mathematical properties. Since it is based on mathematical model, which is rotation and scale invariant (theoretically), in this framework, vector-based symbol recognition can achieve its optimal performance.

Formal learning techniques help characterize 2D graphical symbols [18]. For example, authors introduced the inductive learning programming (ILP) tool to automatically learn symbols that are formally described (first-order logic). An objective of the technique is to describe 2D graphical symbols by the controlled set of visual cues or primitives and their relationships, i.e., spatial relations. Visual cues or primitives can be of any complexity; not necessarily just basic lines and/or points.

In brief, following the state-of-the-art literature, the use of syntactic approaches is limited and is application dependent, since it is not trivial to transform statistical features or signatures into spatial relations (spatial predicates), such as *left* and *right*, and *top*, and *bottom*. If transformed, it can be inaccurate and the semantic description does not contain complete information about the symbol as compared to statistical values. However, such approaches are always appropriate in case it requires close-to human language interpretation/representation.

7.2 Inductive Logic Programming (ILP)

7.2.1 Basics to ILP

ILP [19–21] is a topic built at the intersection of machine learning (ML) and logic programming (LP). In other words, it combines automatic learning and first-order logic (FOL) programming. In the following, we describe FOL and how can we use ILP.

(a) *Machine learning*:
Learning from the experience (data) for any provided task aims at advancing performance [22], which is a research area under artificial intelligence (AI) [23–25]. Like all subjects in AI, machine learning requires interdisciplinary proficiency. Different topics can be enlisted as probability theory, statistics, pattern recognition, data mining, cognitive science, and computational neuroscience. It is primarily focussed to automatically recognize complex patterns and produce intelligent decisions based on experience. However, the difficulty lies in complexity of the problems. This means that there can be several different sets of all possible characteristics from all possible inputs. In one word, intuitive theories are based on human knowledge; meaning, systems of abstract concepts that organize, predict, and describe the observations [26]. It resembles an interaction of human and machine learning.
Machine learning uses several different algorithms, such as decision tree learning (DTL), association rule learning (ARL), artificial neural network (ANN), genetic programming (GP), Bayesian network (BN), support vector machine (SVM), reinforcement learning (RL), and ILP [2]. Among many applications, such as natural language processing (NLP), syntactic pattern recognition (SPR), search engines, medical diagnosis, bioinformatics, classifying DNA sequences, speech processing, and handwriting recognition, in this chapter, we focus on ILP to check and learn whether graphical symbols can be well described. It is important to note that, regardless of the applications, the main aim of the learning is to characterize the experience [27].
Let us discuss FOL first and then ILP.

Table 7.1 Language of FOL—grammar [28]

$$\begin{aligned}
\text{Sentence} &::= \text{AtomicS} \mid \text{ComplexS}\\
\text{AtomicS} &::= \text{True} \mid \text{False} \mid \text{RelationSymb(Term, \ldots)} \mid \text{Term} = \text{Term}\\
\text{ComplexS} &::= \text{(Sentence) Sentence Connective Sentence} \mid \neg\text{Sentence}\\
&\quad \mid \text{Quantifier Sentence}\\
\text{Term} &::= \text{FunctionSymb(Term, \ldots)} \mid \text{ConstantSymb} \mid \text{Variable}\\
\text{Connective} &::= \wedge \mid \vee \mid \Rightarrow \mid \Leftrightarrow\\
\text{Quantifier} &::= \forall\ \text{Variable} \mid \exists\ \text{Variable}\\
\text{Variable} &::= \text{a} \mid \text{b} \mid \ldots \mid \text{x} \mid \text{y} \mid \ldots\\
\text{ConstantSymb} &::= \text{A} \mid \text{B} \mid \ldots \mid \text{John} \mid 0 \mid 1 \mid \ldots \mid \pi \mid \ldots\\
\text{FunctionSymb} &::= \text{F} \mid \text{G} \mid \ldots \mid \text{Cosine} \mid \text{Height} \mid \text{FatherOf} \mid + \mid \ldots\\
\text{RelationSymb} &::= \text{P} \mid \text{Q} \mid \ldots \mid \text{Red} \mid \text{Brother} \mid \text{Apple} \mid > \mid \ldots \mid
\end{aligned}$$

(b) *First-order logic*:
It is close to natural language that considers the problem is composed of *objects*, with individual *identities* and *characteristics* that help distinguish, and *relations* [28]. Since relations can be *functional*, we can consider it as a powerful representation. More often, in FOL, relations (between the objects) are used to build predicates, such as

(a) loved(David, Alex),
(b) above(Ball, Table) and
(c) Russian(David).

In example (a), loved refers to relation between terms or variables: David and Alex. For example (b), an object Ball is above another object called Table. Similarly, an object David is a Russian citizen, expressed in example (c). Following Table 7.1, a very simple semantic of FOL can be expressed as

(∀ y Black(y) ≡ Black(Object1) ∧ Black(Object2)
∧ Black(Object3) ∧...)

Using the same FOL (mentioned earlier), let us have an example (see below):

∀ y (animal(y) ∧ ¬ tiger(y)) ⇒ jumps(y).

7.2.2 How Does ILP Work?

In brief, ILP [20] combines automatic learning and first-order logic programming. It requires three main sets of information, the automatic solving and deduction theory set aside (see Fig. 7.1):

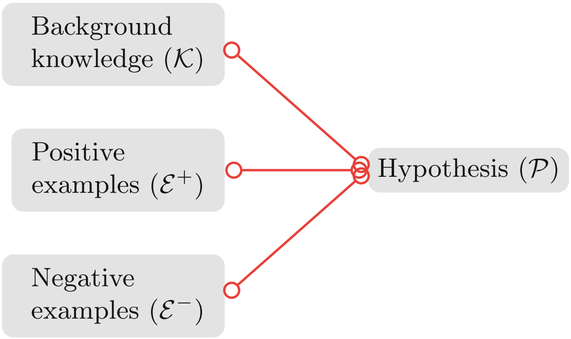

Background knowledge ($\mathcal{K}$)
Positive examples ($\mathcal{E}^+$)
Negative examples ($\mathcal{E}^-$)
Hypothesis ($\mathcal{P}$)

Fig. 7.1 Inductive Logic Programming (ILP) framework

(a) Domain knowledge base $\mathscr{K}$,
(b) A set of positive examples $\mathscr{E}^+$, and
(c) A set of negative examples $\mathscr{E}^-$.

Domain knowledge base $\mathscr{K}$ is composed of a set of known vocabulary, rules, and predicates (axioms). $\mathscr{E}^+$ characterizes with the set of predicates of $\mathscr{K}$ and $\mathscr{E}^-$ aimed to be excluded from the system. With these data, the ILP concludes the set of properties $\mathscr{P}$ in terms of predicates and domain knowledge $\mathscr{K}$ such that $\mathscr{E}^+$ can satisfy $\mathscr{P}$. In document image understanding [29, 30] and to retrieve semantics (from the input texts) [31], such an approach has been widely used. The approach lets us learn common characteristics within classes. This means that it can extract and express nontrivial semantics. In the following, the ILP solver—Aleph—is freely available from the Oxford University Computing Lab[1] will be used.

7.2.3 *ILP for Character/Text Recognition*

ILP has been already been successfully used in many areas. However, this section is mainly related to document analysis and recognition—any type of symbols including handwritten characters. More specifically, we review the way how images and or texts are described.

(a) *Character recognition*
As reported in [29, 30, 32], document image (structure) can be analyzed with the use of the ILP, where it learns letters via structural description. Structural description has been made through the use of visual cues or primitives in addition to their relations (physical). As an example, the description of a letter `E` can be described as shown in Fig. 7.2. In this example, we describe the `letter` via the use of `stroke`, relations, such as `part_of`, `line` and spatial predicates, such as `right top`.

[1] http://web2.comlab.ox.ac.uk/oucl/research/areas/machlearn/Aleph/aleph.html.

Fig. 7.2 Description of a letter (adopted from [32])

`letter(P)`	`P` is a `letter`
`part_of(o,P)`	`o` is a part of `P`
`stroke(l)`	`l` is a `stroke`
`right top(o,l)`	`o` is to the right top of`l`
...	

(b) *Semantic extraction*
By extracting semantics from written texts [31, 33], complex queries can be managed. It relies on a simple but through the use of relations. To understand this, a typical example of how we can learn logic programs (see Fig. 7.3) is important. The idea is to represent a problem by the defined first-order Horn-clause (dependency tree see Fig. 7.3, adopted from [33]). Let us discuss more about how relational structure can be expressed (see below):

```
is(a1). Fraun(a2). Optician(a3). a(a4). German(a5).
```

In general, use of the relational operator, R can be used: R`(a2,a1)`, for instance. With this, the aforementioned example can be encoded by `is_a(Fraun,Optician)`.

(c) *Information extraction*
To extract relevant information from unstructured texts [34], ILP has been an attraction for several years. For example, in message understanding and information extraction, objects/entities, such as person's name, addresses, and affiliations, are widely terms [35, 36]. Further, use of the experts is important. In case of named entity recognition (NER), experts (linguistic) have to be integrated in addition to the use of the ILP [34]. Let us remind that ILP does not substitute linguists, but it can use as a complementary tools where automation is required. Based on Fig. 7.4, background knowledge can be written as

b_entity`(Name).`	b_entity`(Organization).`	b_entity`(Address).`	...
b_word`(Mr.).`	b_word`(KC).`	b_word`(lives).`	...
b_tag`(Verb).`	b_tag`(Noun).`	...	...
...			

(d) *Semantic distribution*
The probabilistic logical model has been recently used [37–39] and for example, first-order logical probability tree is shown in Fig. 7.5. In Fig. 7.5, `student(S)` and `course(C)` as well as simple probability FOL predicates are used to describe.
With this formal description (for any problem), ILP works on reasoning.

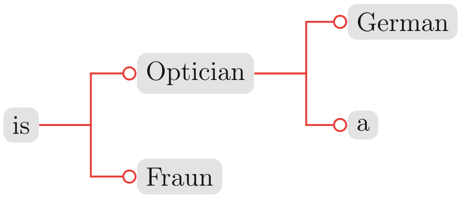


Fig. 7.3 Dependency tree: "Fraun is a German Optician"

Fig. 7.4 Entities, an example

Name	Mr. KC described her ideas.
Address	He lives in DC.
Affiliation	He teaches Computer Science.
...	

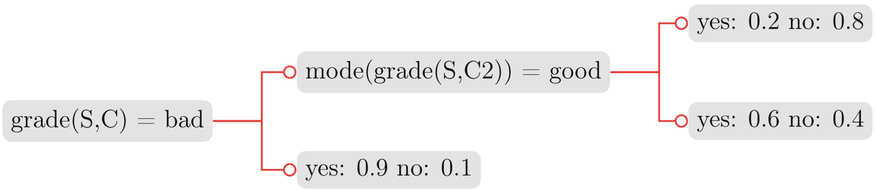


Fig. 7.5 First-order logical probability tree [37]

7.3 ILP for Graphical Symbol Recognition

7.3.1 Overview

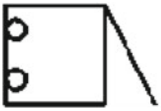
Bridging the semantic gap between low-level image description and the content within the image is important [2, 18]. As an example, it is assumed that patterns can be described via their shape (shape feature) so that the comparison can be made to perform classification or clustering. Expressing information contents that are close-to human description or "natural" description has been always an interesting research topic, which is in the framework of structural pattern recognition. In the framework of structural approaches, low-level visual cues or primitives that helps build lexical data is the first step terminal concepts, where spatial predicates are widely used so that higher level description (rules) can be possible [40].

In this approach, we aim to describe visual cues primitives (structures, for instance) that compose a complete symbol. As an example, few visual primitive types are considered (Fig. 5.2 of Chap. 5). Once the 2D graphical symbol is expressed as a set of visual primitives, their relations can be computed using the quantitative assessment of directional spatial relationships, such as "to the right of," "above," and "south of." To compute spatial predicates, the vanilla version of the force histogram-based approach was used [41–43]. This helps us describe a symbol with FOL predicates, visual cues/primitives, and their possible relations (relative positions). In general, such a framework yields an interesting way to describe any studied image. It contains both expressiveness and flexibility. However, the actual set of visual cues can be small and/or large and therefore, their relations. As an example, we are able to reduce or extend the size in accordance with the need and the algorithms to describe relations. For example, the use of statistical- or signal-based extractions can help optimize the sie [44]. Let us remind that relations that convey the relative positioning between the visual cues can also include quantitative information (e.g., [45–47]).

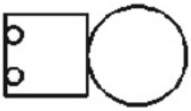
7.3.2 Graphical Symbol Representation

Let us follow exactly similar procedure to extract visual primitives from the graphical symbol (see Chaps. 4–6). As a reminder, in Fig. 7.6, few sample images and the corresponding visual primitives are shown. It is important to note that we keep their spatial positions as they are in the original image. In our illustration, for symbol 141_2, there are three types of visual primitives. Symbol 180_3 is composed of four types of visual primitives. In a similar fashion, there are only two types of visual primitives (*circle* and *corner*) that compose a symbol 226_2.

For a better understanding, let us take two sample images (see below):



a) symbol 225_2



b) symbol 226_2

Both of them can be described as follows (adopted from [2]):

```
 % start: symbol 225_2*****************************
type(primitive_170,cornerne). type(primitive_171,cornernw).
type(primitive_172,cornerse). type(primitive_173,extremity).
has(img_225_2,primitive_170).
has(img_225_2,primitive_171).
has(img_225_2,primitive_172).
has(img_225_2,primitive_173).
nw(primitive_170,primitive_171).
n(primitive_170,primitive_172).
nw(primitive_170,primitive_173).
se(primitive_171,primitive_170).
ne(primitive_171,primitive_172).
n(primitive_171,primitive_173).
s(primitive_172,primitive_170).
sw(primitive_172,primitive_171).
w(primitive_172,primitive_173).
se(primitive_173,primitive_170).
s(primitive_173,primitive_171).
```

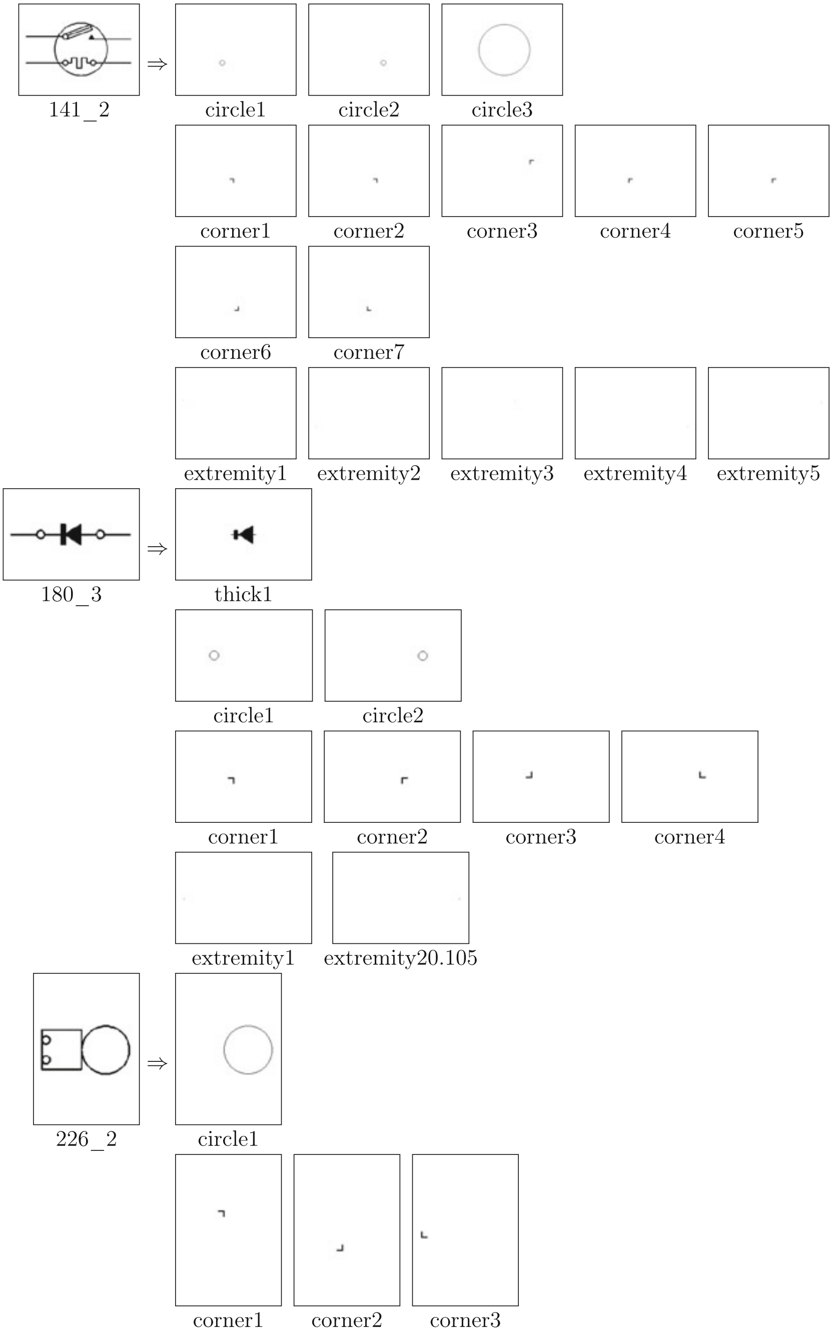


Fig. 7.6 Symbol decomposition via visual cues or primitives: *thick*, *circle*, *corner*, and *extremity* (adopted from [2])

```
e(primitive_173,primitive_172).
% end: symbol 225_2*****************************

 % start: symbol 226_2*****************************
type(primitive_174,circle).
type(primitive_175,cornerne).
type(primitive_176,cornerse).
type(primitive_177,cornersw).
has(img_226_2,primitive_174).
has(img_226_2,primitive_175).
has(img_226_2,primitive_176).
has(img_226_2,primitive_177).
e(primitive_174,primitive_175).
e(primitive_174,primitive_176).
inside(primitive_174,primitive_177).
w(primitive_175,primitive_174).
n(primitive_175,primitive_176).
nw(primitive_175,primitive_177).
w(primitive_176,primitive_174).
s(primitive_176,primitive_175).
w(primitive_176,primitive_177).
inside(primitive_177,primitive_174).
se(primitive_177,primitive_175).
e(primitive_177,primitive_176).
% end: symbol 226_2*****************************
```

With such a description (symbol 225_2, for instance), we observe that the first two lines refer to the visual primitive types:

```
type(primitive_AA, visual_primitive).
```

After that, the next four lines tell us where they come from, i.e., `image_name`

```
has(image_name,primitive_AA).
```

The last six lines convey the pairwise spatial relations using the spatial predicates,

```
s(primitive_AA,primitive_BB)
```

i.e., `primitive_AA` is to the *south* `primitive_BB`.

7.3.3 Graphical Symbol Recognition

The primary idea is to learn common characteristics from set of chosen symbols. This helps express nontrivial knowledge of visual representations, relying on semantic concept. Let us have a few tests such that we can understand how ILP solver works.

(a) *Proof of a concept, example*:
To prove the concept, let us take a small set of symbols, as shown in Fig. 7.7. From this set, to show what kind of data we actually manipulate, let us select two symbols (named 225_2 and 226_2) as positive examples. The remaining symbols are taken as negative examples, i.e., counterexamples.
At the output (via ILP solver), it has a `[theory]` section. We expect to have rules in it and they are related to positive examples. For each rule, the solver gives matching statistics. It indicates how precise are the rules. The rule is said to be perfect if it produces, in the theory section, one single rule. Such a rule covers all positive examples, where no negative examples are covered. The latter case is often ideal. More often, the theory covers multiple rules. Each of them covers a subset of the positive examples. Further, it is not surprising to see negative examples in the theory. But, in this example, we have

```
[theory]
[Rule 1] [Pos cover = 2 Neg cover = 0]
symbol(X):-
    has(X,Y), type(Y,cornerne),
    has(X,Z), n(Y,Z), type(Z,cornerse).

[positive examples covered]
    symbol(img_225_2).
    symbol(img_226_2).
[negative examples covered]

test
[covered]
symbol(img_225_2):-
     has(img_225_2,primitive_170),
     type(primitive_170,cornerne),
     has(img_225_2,primitive_172),
     n(primitive_170,primitive_172),type(primitive_172,
cornerse).
```

The last part of the theory `[covered]` provides an example of one symbol used (from a set of positive examples). This allows us to have “visual” verification, for better understanding. In our example, the complete interpretation of the output of our solver is that symbols 225_2 and 226_2 can be formally and completely distinguished from other counter symbols using two corners in the following positions:

┓ *north-east* corner
┛ *south-east corner*

Note that, if we check counterexamples in Fig. 7.7, we do not find exactly similar spatial positioning of two corners (as shown above).

(b) *Global behavior, extension*:
Since aforementioned example (set of symbols) looks easy to characterize positive samples with respect to counter (negative) samples, we better move forward by taking more difficult scenario. "Easy to characterize" refers to trivial solution (rule) from ILP. For new tests, we still consider exactly same set as shown in Fig. 7.7.

- Let us consider symbols {195_2, 198_2, 199_2, 200_2, 207_2, 208_2} as positive examples, from which we expect to induce theory. Other symbols are taken as counter examples. Below is the output from the ILP solver:

```
symbol(X):-
   has(X,Y), type(Y,circle),
   has(X,Z), inside(Z,Y),
   type(Z,cornernw).
```

  In this test, the chosen positive examples have circles containing a visual primitive type, *northwest* corner ┏. We observe that all positive examples are characterized with the rule as mentioned before.
- In case of another set of positive examples, i.e., symbols, such as {180_1, 180_3, 184_1, 185_1, 185_3, and 186_2}, the ILP solver produces multiple rules (see below).

```
[Rule 1] [Pos cover = 1 Neg cover = 0]
symbol(img_180_1).

[Rule 2] [Pos cover = 2 Neg cover = 0]
symbol(X):-
   has(X,Y), type(Y,circle),
   has(X,Z), e(Y,Z),
   type(Z,cornernw).

[Rule 3] [Pos cover = 3 Neg cover = 0]
symbol(X):-
   has(X,Y), type(Y,blackthick),
   has(X,Z), type(Z,cornersw),
   has(X,A), ne(Z,A).
```

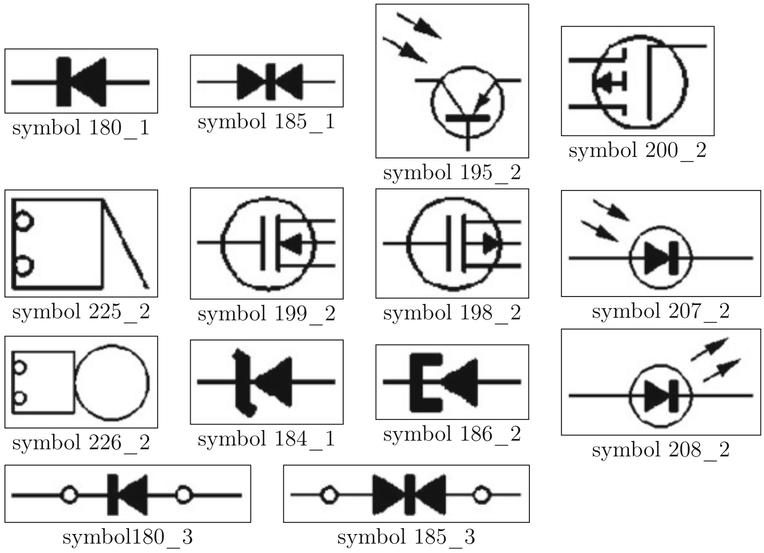


Fig. 7.7 A small set of symbol

In this test, symbol 180_1 is not covered and for the remaining positive symbols, multiple rules are produced, where each of them covered by a separate rule.

(a) `[Rule 2]` describes the symbols that has a circle ○ and a corner ┏; and circle is to the east of corner.

(a) In `[Rule 3]`, we observe that the corresponding symbols has visual primitives: *thick* ◀ and corner └. Surprisingly, it has a third but *unspecified*, primitive, which is at the *northeast* location with respect to the defined corner.
We have learned that ILP solver can induce generic relationships (regardless of the underlying shape!) in contrast to quantitative classification [48, 49].

7.4 Summary

In this chapter, we have discussed the first step toward another approach of graphics recognition by taking visual primitives and spatial predicates. In other words, like in structural approaches, we have taken spatial predicates to take visual primitives into account so that FOL-based description of the symbol is possible. Further, ILP can be used to extract "semantic" contexts or concepts from a set of graphical symbols.

The interesting part of such an approach is that the description of the symbols can now be integrated/combined with other, more context-related information. Let us summarize the approach in just two major points:

1. The information need not necessarily be visually represented (for example, from surrounding text). As a consequence, it opens a new scope of possible combined text/image concept characterization and learning. It is also possible to extend such a concept to generate symbols from the FOL descriptions (for visual validation of classification results, for instance) [50].
2. In contrast to statistical models, it can adapt the complexity of the classification with respect to the learning data, where parameters are not required.
   Besides, it can induce generic relationships (regardless of the underlying shape!) in contrast to quantitative classification [48, 49]. On the other hand, it is observed that if the learning set is contradictory, it cannot produce rule that helps classify the set of symbols. An important point is that the method, as it currently stands, depends on the visual cues/primitives.

As we observed that the performance and its extensions are based on what we have extracted, i.e., visual primitives that compose the symbol, the idea can be extended by increasing the visual cues. Besides, refining spatial predicates can help in precision. This means that use of relative distance and size, such as close, far, large, and small can strengthen the image description. We are aware of that extensions can be possible, since state-of-the-art image analysis can be used in accordance with the need. More information can be found in [2, 18]. At the same time, considering statistical descriptions can be a better quantifier. For example, the difference between the shapes that relies on Markov logic can handle precise values (statistical) [51, 52]. Regarding graphical symbol recognition, a few specific reported works can be [53–58], where relational signatures are used instead of relying on high-level semantics (i.e., predicates, such as *Left* and *Right*). Not a surprising, these techniques are taken from the statistical and structural approaches (previous chapters).

Prospective works will be to combine such a concept to formal concept analysis [59] and Galois lattices to achieve unsupervised learning of visual concepts. Further, the use of bag-of-relations for recognition as discussed and validated in Chap. 5 can be taken as one of many interesting ideas.

References

1. Sebastian Lutz. What's right with a syntactic approach to theories and models? *Erkenntnis*, pages 1–18, 2014
2. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis). Ph.D. thesis, University of Lorraine, France (2011)
3. W.H. Tsai, K.S. Fu, Attributed grammar: a tool for combining syntactic and statistical approaches to pattern recognition. IEEE Trans. Syst. Man Cybern. **10**(12), 873–885 (1980)

4. M. Viswanathan, Analysis of scanned documents - a syntactic approach, in *Structured Document Image Analysis*, ed. by H.S. Baird, H. Bunke, K. Yamamoto (Springer, Heidelberg, 1992), pp. 115–136
5. A. Rosenfeld, Array, tree and graph grammars, in *Syntactic and Structural Pattern Recognition: Theory and Applications (Chap. 4)*, ed. by H. Bunke, A. Sanfeliu (World Scientific, Singapore, 1990), pp. 85–115
6. H. Fahmy, D. Blostein, A survey of graph grammars: theory and applications, in *International Conference on Pattern Recognition, Vol.II. Conference B: Pattern Recognition Methodology and Systems* (1992), pp. 294–298
7. D. Dori, A. Pnueli, The grammar of dimensions in machine drawings. Comput. Vis. Graph. Image Process. **42**, 1–18 (1988)
8. D. Dori, Dimensioning analysis: toward automatic understanding of engineering drawings. Commun. ACM **35**(10), 92–103 (1992)
9. S. Collin, D. Colnet, Syntactic analysis of technical drawing dimensions. Int. J. Pattern Recognit. Artif. Intell. **8**(5), 1131–1148 (1994)
10. T. Feder, Plex languages. Inf. Sci. **3**, 225–241 (1971)
11. H. Bunke, Attributed programmed graph grammars and their application to schematic diagram interpretation. IEEE Trans. Pattern Anal. Mach. Intell. **4**(6), 574–582 (1982)
12. S.H. Joseph, T.P. Pridmore, Knowledge-directed interpretation of mechanical engineering drawings. IEEE Trans. Pattern Anal. Mach. Intell. **14**(9), 928–940 (1992)
13. G. Sánchez, J. Lladós, A graph grammar to recognize textured symbols, in *Proceedings of the 6th International Conference on Document Analysis and Recognition, Seattle, WA (USA)* (2001), pp. 465–469
14. G. Sánchez, J. Lladós, K. Tombre, A mean string algorithm to compute the average among a set of 2D shapes. Pattern Recognit. Lett. **23**(1), 203–213 (2002)
15. J. Song, F. Su, C.-L. Tai, S. Cai, An object-oriented progressive-simplification based vectorization system for engineering drawings: model, algorithm, and performance. IEEE Trans. Pattern Anal. Mach. Intell. **24**(8), 1048–1060 (2002)
16. J. Lladós, G. Sánchez, Graph matching versus graph parsing in graphics recognition - a combined approach. Int. J. Pattern Recognit. Artif. Intell. **18**(3), 455–473 (2004)
17. Y. Yajie, W. Zhang, L. Wenyin, A new syntactic approach to graphic symbol recognition. Proc. Int. Conf. Doc. Anal. Recognit. **1**, 516–520 (2007)
18. K.C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, 2009), pp. 1330–1334
19. G.D. Plotkin, Automatic methods of inductive inference. Ph.D. thesis, Edinburgh University (1971)
20. S. Muggleton, L. De Raedt, Theory and methods. Inductive logic programming. J. Logic Progr. **19**, 629–679 (1994)
21. S.-H. Nienhuys-Cheng, R. de Wolf, *Foundations of Inductive Logic Programming* (Springer, New York, 1997)
22. T.M. Mitchell, *Machine Learning* (McGraw Hill, New York, 1997)
23. R.S. Michalski, J.G. Carbonell, T.M. Mitchell, *Machine Learning: An Artificial Intelligence Approach*, 2nd edn. (Morgan Kaufmann, Los Altos, 1986)
24. Y. Kodratoff, R.S. Michalski, *Machine Learning: An Artificial Intelligence Approach*, 3rd edn. (Morgan Kaufmann, San Mateo, 1990)
25. R.S. Michalski, G. Tecuci, *Machine Learning: A Multistrategy Approach*, 4th edn. (Morgan Kaufmann, San Francisco, 1994)
26. J.B. Tenenbaum, Building theories of the world: Human and machine learning perspectives, *ILP*. Lecture Notes in Computer Science, vol. 5194 (Springer, Berlin, 2008), p. 1
27. C.M. Bishop, *Pattern Recognition and Machine Learning* (Springer, Berlin, 2006)
28. S. Russell, P. Norvig, *Artificial Intelligence: A Modern Approach*, 3rd edn. (Prentice Hall, Upper Saddle River, 2010)

29. A. Amin, C. Sammut, K.C. Sum, Learning to recognize hand-printed Chinese characters using inductive logic programming. Int. J. Pattern Recognit. Artif. Intell. **10**(7), 829–847 (1996)
30. M. Ceci, M. Berardi, D. Malerba, Relational data mining and ilp for document image processing. Appl. Artif. Intell. **21**(8), 317–342 (2007)
31. V. Claveau, P. Sébillot, From efficiency to portability: acquisition of semantic relations by semi-supervised machine learning, in *Proceedings of International Conference on Computational Linguistics, Geneva, Switzerland* (2004), pp. 261–267
32. A. Amin, Recognition of hand-printed characters based on structural description and inductive logic programming. Pattern Recognit. Lett. **24**(16), 3187–3196 (2003)
33. T. Horváth, G. Paass, F. Reichartz, S. Wrobel, A logic-based approach to relation extraction from texts, *ILP*. Lecture Notes in Computer Science, vol. 5989 (Springer, Berlin, 2009), pp. 34–48
34. A. Patel, G. Ramakrishnan, P. Bhattacharya, Incorporating linguistic expertise using ilp for named entity recognition in data hungry indian languages, in *Proceedings of the 19th international Conference on Inductive logic programming* (Springer, Berlin, 2010), pp. 178–185
35. R. Grishman, B. Sundheim, Message understanding conference - 6: a brief history, in *Proceedings of the International Conference on Computational Linguistics* (1996)
36. L. Song, X. Cheng, Y. Guo, Y. Liu, G. Ding, Contentex: a framework for automatic content extraction programs, in *IEEE International Conference on Intelligence and Security Informatics* (2009), pp. 188 –190
37. D. Fierens, On the relationship between logical bayesian networks and probabilistic logic programming based on the distribution semantics, in *Proceedings of the 19th International Conference on Inductive Logic Programming* (Springer, Berlin, 2010), pp. 17–24
38. L. De Raedt, K. Kersting, *Probabilistic inductive logic programming, Probabilistic Inductive Logic Programming*, vol. 4911 (Springer, Berlin, 2008), pp. 1–27
39. L. Getoor, B. Taskar, *Introduction to Statistical Relational Learning* (MIT Press, Cambridge, 2007)
40. J.M. Romeu, G. Sanchez, J. Llados, B. Lamiroy, An incremental on-line parsing algorithm for recognizing sketching diagrams, in *Proceedings of International Conference on Document Analysis and Recognition, Curitiba Brasil*, ed. by F. Bortolozzi, R. Sabourin (2007), pp. 452–456
41. P. Matsakis, L. Wendling, A new way to represent the relative position between areal objects. IEEE Trans. Pattern Anal. Mach. Intell. **21**(7), 634–643 (1999)
42. P. Matsakis, J. Keller, L. Wendling, J. Marjamaa, O. Sjahputera, Linguistic description of relative positions in images. IEEE Trans. Syst. Man Cybern.- Part B Cybern. **31**(4), 573–588 (2001)
43. P. Matsakis, J.M. Keller, O. Sjahputera, J. Marjamaa, The use of force histograms for affine-invariant relative position description. IEEE Trans. Pattern Anal. Mach. Intell. **26**(1), 1–18 (2004)
44. T.-O. Nguyen, S. Tabbone, O.R. Terrades, Symbol descriptor based on shape context and vector model of information retrieval, in *Proceedings of International Workshop on Document Analysis Systems* (2008), pp. 191–197
45. I. Bloch, Fuzzy spatial relationships for image processing and interpretation: a review. Image Vis. Comput. **23**, 99–110 (2005)
46. B. Bennett, P. Agarwal, Semantic categories underlying the meaning of ‘place’, *Proceedings of International Conference on Spatial Information Theory*. Lecture Notes in Computer Science, vol. 4746 (Springer, Berlin, 2007)
47. K.C. Santosh, L. Wendling, B. Lamiroy, New ways to handle spatial relations through angle plus mbr theory on raster documents, in *Proceedings of IAPR International Workshop on Graphics Recognition, La Rochelle, France* (2009), pp. 291–302
48. C. Malon, S. Uchida, M. Suzuki, Mathematical symbol recognition with support vector machines. Pattern Recognit. Lett. **29**(9), 1326–1332 (2008)
49. L.I. Kuncheva, Diversity in multiple classifier systems. Inf. Fusion **6**(1), 3–4 (2005)

50. B. Lamiroy, K. Langa, B. Leoutre, Assessing classification quality by image synthesis, in *Proceedings of IAPR International Workshop on Graphics Recognition, La Rochelle, France* (2009)
51. M. Richardson, P. Domingos, Markov logic networks. Mach. Learn. **62**, 107–136 (2006)
52. P. Domingos, S. Kok, D. Lowd, H. Poon, M. Richardson, P. Singla, *Markov logic, in Probabilistic Inductive Logic Programming* (Springer, Berlin, 2008), pp. 92–117
53. K.C. Santosh, L. Wendling, B. Lamiroy, Unified pairwise spatial relations: an application to graphical symbol retrieval, in *Proceedings of IAPR International Workshop on Graphics Recognition* (2009), pp. 163–174
54. K.C. Santosh, L. Wendling, B. Lamiroy, Using spatial relations for graphical symbol description, in *Proceedings of the IAPR International Conference on Pattern Recognition* (2010), pp. 2041–2044
55. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
56. K.C. Santosh, B. Lamiroy, L. Wendling, Spatio-structural symbol description with statistical feature add-on, in *Graphics Recognition. New Trends and Challenges*, vol. 7423, Lecture Notes in Computer Science, ed. by Y.-B. Kwon, J.-M. Ogier (Springer, Berlin, 2011), pp. 228–237
57. K.C. Santosh, B. Lamiroy, L. Wendling, Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Doc. Anal. Recognit. **17**(1), 61–78 (2014)
58. K.C. Santosh, L. Wendling, B. Lamiroy. Relation bag-of-features for symbol retrieval. In *12th International Conference on Document Analysis and Recognition*, pages 768–772, 2013
59. Gerd Stumme Bernhard Ganter and Rudolf Wille (eds.), editors. *Formal Concept Analysis: Foundations and Applications*. Number 3626. Springer-Verlag, 2005

Chapter 8
Conclusion and Challenges

8.1 Summary State-of-the-Art Works and Extensions

We have discussed a research topic: graphical symbol recognition, which is considered as a challenging subfield of the research domain: pattern recognition (PR). Within the PR framework, it has been taken as a key task toward document content understanding and interpretation, and mostly architectural, engineering drawings, and elecDBLP:phd/hal/Santosh11atrical circuit diagrams. In brief, starting with its definition, the book discussed basic steps that are taken from the state-of-the-art methods, a few projects, and key research standpoints. Specifically, research standpoints are relying on the state-of-the-art works that were addressing graphics recognition [1]. For a clear and concise report, readers can take a note/message reported work [2].

At the time (around 60 and 70 s) when the resource-constrained machines did not allow complex data representation and/or recognition techniques [3], it was difficult to automate a tool that has to be dealt with big data. With the increasing demand and the evolution of more powerful machines, interactions between disciplines and new projects on data mining, document taxonomy led the progress in many ways or concepts [4]. Since the 70s, graphics recognition has a rich state-of-the-art literature [5, 6]. In the literature, the state-of-art works are grouped into the three different categories/approaches: statistical, structural, and syntactic.

In all cases (approaches, mentioned earlier), the methods have been tested in accordance with the context, i.e., defined problem that may be restricted by the industrial needs, for instance, and the provided dataset. Within this framework, the recognition problem is trivial, where two (test and model) symbols are aligned/matched to check how similar they are. The similarity, more often, relies on the computed distance between the features representing the patterns. The test symbol is said to be correctly classified as the model symbol or class from which it yields the highest similarity score. As an extension, for a retrieval task, methods are able to short-list model symbols in accordance with the order of similarity. Other methods are positioned with different applications, where the recognition of graphical elements

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3_8

and/or the localization of significant or known visual parts are crucial. The latter work is referred to as symbol spotting. Symbol spotting basically user-driven, where test query can be either an isolated graphical symbol or other graphical elements (meaningful parts) that signify the common characteristics of a set of train symbols (*Ref.* Chap. 2). For evaluation, we have observed that recognition rate (accuracy), precision and recall, F-measure, ROC curve, and confusion matrices are common performance measures. It is important to note that computing the aforementioned metrics is not obvious since ground truths are uncertain and missed in case of real-world data [7]. Therefore, for such a situation, as an alternative solution, retrieval efficiency can be taken as a retrieval quality measure in case of datasets, where the number of similar symbols varies from one class to another (imbalanced but labeled ground truths). Not a surprising, it often happens in real-world project [1]. Several different techniques/approaches are found in the literature. As stated earlier, two major points: datasets and evaluation metrics, are important to make a fair comparison. This means that, in order to see, how far we have been advanced, one needs to follow the exact similar evaluation protocol. More often, the characteristics of the datasets, their availability for further researches, and the applications (or intentions) may change one's evaluation metric. Besides, one may be biased in re-implementing previously reported algorithms/techniques. As a consequence, we are unable to track researches done over several years, since results cannot be consistent as algorithms may not be tuned (i.e., parameters) as in the original references [8]. As reported in [9], document analysis and exploitation (DAE) was conceived and built around a core data model that establishes an exhaustive range of relations between document images, annotation areas, interpretations, or ground truth. It also connects the data to user interactions, experimental protocols, or program executions. In Chap. 3, more detailed discussion has been made on several different services, such as querying, up- and download, and remote execution.

Based on our review, statistical approaches are appropriate to recognize isolated symbols as they are robust to noise (of almost all types), degradations, deformations, and occlusions. Statistical signatures (shape-based signatures) are basically simple (1D feature vector) to compute with low computational cost. Several different signal-based features can be combined. Discrimination power and robustness, however, strongly depend on the selection of an optimal set of features. Integrating features are not straightforward and trivial, since appropriate fusion of classifiers is also crucial. A more detailed information can be taken from Chap. 4 and [?] for extended results.

On contrary, structural approaches are particularly well suited for recognizing complex and composite graphical symbols (*Ref.* Chap. 5 and previous works [10, 11]). Under this framework, graphical elements/symbols can be used for spotting/localization. For example, these techniques/algorithms are designed to recognize meaningful region-of-interest that can be a complete graphical symbol or any basic shapes representing the characteristics of any particular graphical symbol in technical documents. In structural approaches, methods are relying on symbolic data structures, such as graphs, strings, and trees. In the state-of-the-art literature, graph-based pattern representation (including matching) has been considered as a prominent technique even if it suffers from high computational cost. Graph matching cost, i.e., computational complexity often increases when complex and composite symbols are taken for study due to the well-known problem: subgraph isomorphism. Further, due to the presence of noise and possible distortions in the studied patterns, graph sizes vary a lot. This variation is taken as one of the reasons that helps increase graph matching computational cost. In contrast to statistical approaches, structural approaches provide a powerful representation since they convey how parts are connected to each other. Such a representation preserves the technique's generality and extensibility. The term "extensibility" allows us to combine/integrate to other methods that come from different approaches.

Since not a single method (either from statistical or structural) provides a satisfactory performance, hybrid approaches (*Ref.* Chap. 6) are designed to check whether they can compliment each other. In other words, hybrid approaches try to integrate best of the two worlds: statistical and structural, for instance. In the previously reported work [?], results have been extended/advanced. Such approaches are often dedicated to the graphical symbol localization in accordance with the specific rules and are based on a set of arbitrary graphical symbols. Not to be confused, the concept of integrating descriptors and classifiers can be different than hybrid approaches. Within the framework, in visual cues/primitive selection, error-prone raster-to-vector conversion can limit the number of applications. As we are aware that primitive extraction is not generic, one can focus on those primitives that are important in that particular application. Therefore, depending on the studied samples, graphs vary. For example, graph can be either proximity graph or line graph. We observed that, often, proximity graph uses local interest points (via computer vision local descriptors) and line graph uses lines (high-level information). Researchers have shown that the line graph is appropriate for technical line drawings.

Syntactic approaches (*Ref.* Chap. 7) describe graphical symbols (or technical documents) using well-mastered grammars (rule-set, for instance). For syntactic approaches, one can use similar primitives as in structural approaches. An idea to use syntactic approaches is to make image description close to the language (first-order logic description). As reported in [12], statistical signatures to spatial predicates conversion may not carry precise information. This means that no metrical details can be found. This results syntactic approaches do not possess detailed information and the approaches may not handle complex and composite documents.

Even though we have not observed that state-of-the-art methods are generic, applications in graphical symbol recognition are not limited. Other than conventional graphics recognition tasks, arrow detection can be considered as one of the graphical symbol/elements and has several different applications. Arrow detection was initially designed for a technical document understanding, where detecting arrows (pointers, in general) can help locate quotation, measurements, and of course, meaningful regions/parts [13–15]. Figure 8.1 shows an example of it. Not a surprising, use of arrow detection can be extended to other domains as well. Arrow detection has recently been considered as an important step in biomedical images to advance the CBIR problem [16–19]. Regardless of applications, often, they aim to address regions-of-interest. Like in technical drawing, detecting overlaid arrows in medical

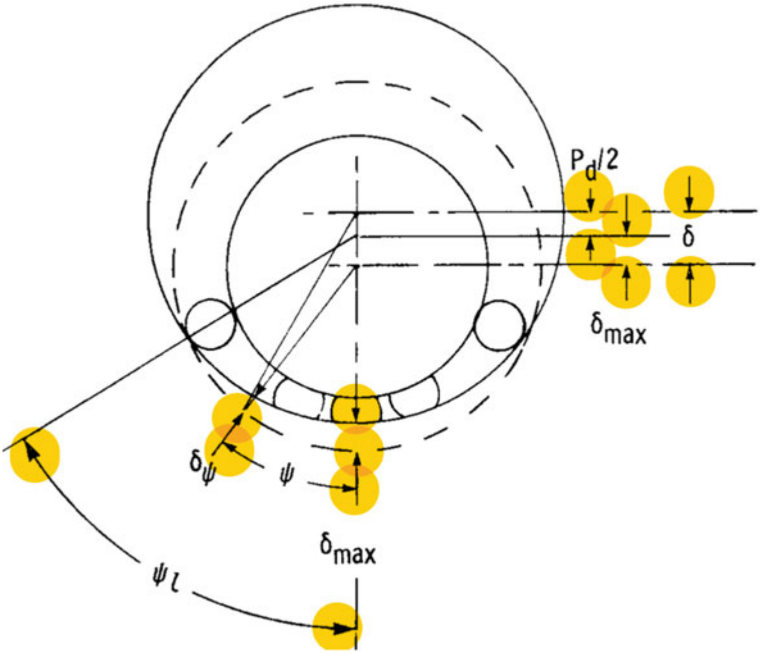


Fig. 8.1 Arrow detection: another important task in graphics recognition. Arrow detection helps locate important quotations and meaningful parts. Highlighted regions (in yellow) are the detected arrows

images can help speed up in region labeling since biomedical images, by nature, tend to be very complex. Few examples are shown in Fig. 8.2. For better understanding, a complete project is demonstrated in Fig. 8.3, where a project from the US National Library of Medicine's (NLM's) entitled Open-i^{SM} image retrieval search engine is provided. In brief, pointer detection can minimize the distractions from other image regions, and more importantly, meaningful regions (regions-of-interest) are often referred to article text and figure captions. It can thus help better analyze the content using other text semantics through the use of natural language processing. Further, can we use pointer location to learn regions-of-interest so that one does not require to learn all pixels (end-to-end) from the image (see Fig. 8.2)? In Fig. 8.2 (right), pointers help learn "infiltrate" without considering all pixels into account. From the machine intelligence (machine learning) viewpoint, one should not stop learning, since learning helps machine robust. This may sometime confuse decisions. Can we just avoid redundancies (via the use of pointer location) from which machines are confused? Of course, let us examine more and extend graphics recognition techniques to another level. In a similar fashion, robust circle-like element detection can help advance abnormality chest X-ray screening [20–22]. These examples can prove that graphics recognition is not just limited to technical drawings, architectural drawings, electrical

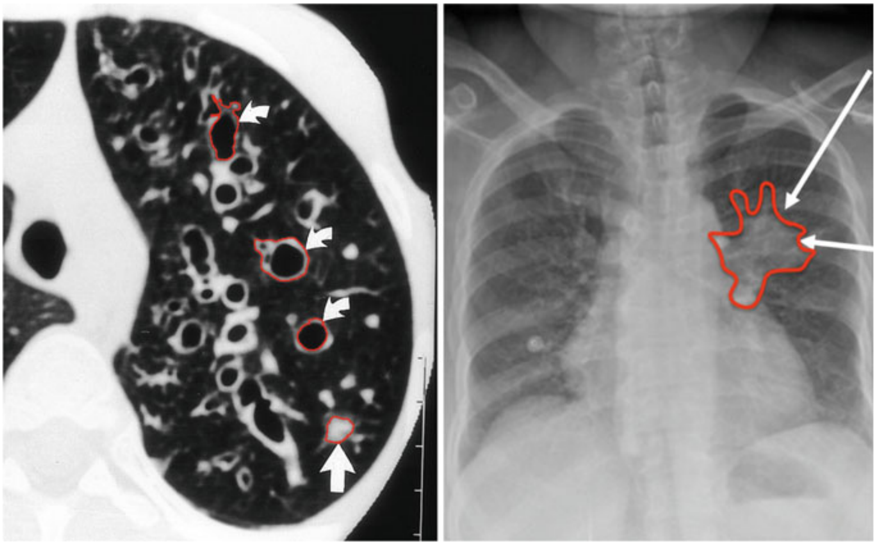
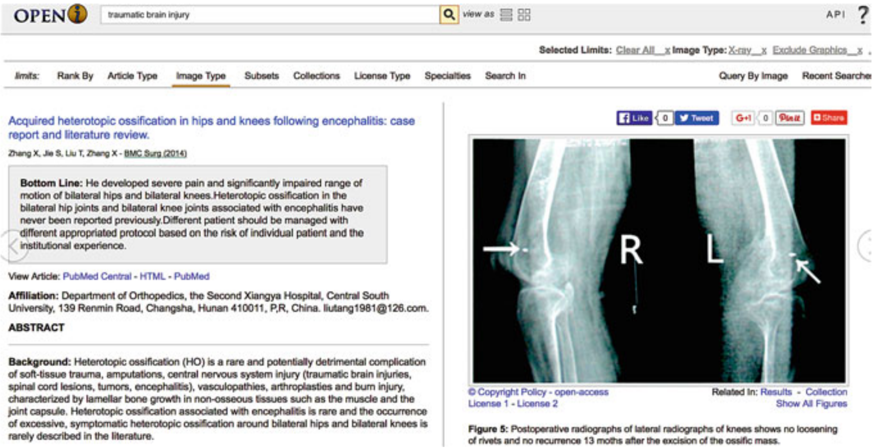


Fig. 8.2 Illustrating the importance of arrow/pointer detection that helps locate meaningful regions. Regions (in red) are labeled as soon as we detect arrows. These regions-of-interest (in red) are automatically generated regions based on the changes in gradients (not annotated by experts)



Acquired heterotopic ossification in hips and knees following encephalitis: case report and literature review.

Zhang X, Jie S, Liu T, Zhang X - BMC Surg (2014)

Bottom Line: He developed severe pain and significantly impaired range of motion of bilateral hips and bilateral knees.Heterotopic ossification in the bilateral hip joints and bilateral knee joints associated with encephalitis have never been reported previously.Different patient should be managed with different appropriated protocol based on the risk of individual patient and the institutional experience.

View Article: PubMed Central - HTML - PubMed

Affiliation: Department of Orthopedics, the Second Xiangya Hospital, Central South University, 139 Renmin Road, Changsha, Hunan 410011, P,R, China. liutang1981@126.com.

ABSTRACT

Background: Heterotopic ossification (HO) is a rare and potentially detrimental complication of soft-tissue trauma, amputations, central nervous system injury (traumatic brain injuries, spinal cord lesions, tumors, encephalitis), vasculopathies, arthroplasties and burn injury, characterized by lamellar bone growth in non-osseous tissues such as the muscle and the joint capsule. Heterotopic ossification associated with encephalitis is rare and the occurrence of excessive, symptomatic heterotopic ossification around bilateral hips and bilateral knees is rarely described in the literature.

© Copyright Policy - open-access
License 1 - License 2

Related In: Results - Collection
Show All Figures

Figure 5: Postoperative radiographs of lateral radiographs of knees shows no loosening of rivets and no recurrence 13 moths after the excision of the ossific mass.

Fig. 8.3 Addressing the usefulness of the annotated arrow in biomedical images. Its location pointing region-of-interest (ROI) and relationship between the texts and ROI (source: US National Library of Medicine's (NLM's) Open-i^{SM} can help advance image retrieval search engine (url: https://openi.nlm.nih.gov))

circuit diagrams, and other business document imaging; it can attract a large audience (up to the level of medical imaging [23]).

References

1. K.C. Santosh, Reconnaissance graphique en utilisant les relations spatiales et analyse de la forme. (Graphics Recognition using Spatial Relations and Shape Analysis). Ph.D. thesis, University of Lorraine, France, 2011
2. K.C. Santosh, L. Wendling, *Graphical Symbol Recognition* (Wiley, New York, 2015), pp. 1–22
3. G. Nagy, State of the art in pattern recognition. Proc. IEEE **56**(5), 836–862 (1968)
4. A.K. Jain, R.P.W. Duin, J. Mao, Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. **22**(1), 4–37 (2000)
5. H. Bunke, P.S.P. Wang (eds.), *Handbook of Character Recognition and Document Image Analysis* (World Scientific, Singapore, 1997)
6. D. Doermann, K. Tombre, *Handbook of Document Image Processing and Recognition* (Springer, New York Incorporated, 2014)
7. B. Lamiroy, T. Sun, Computing precision and recall with missing or uncertain ground truth, in *Graphics Recognition. New Trends and Challenge*, ed. by Y.-B. Kwon, J.-M. Ogier, Lecture Notes, in Computer Science, vol. 7423, (Springer, Berlin, 2013), pp. 149–162
8. B. Lamiroy, D.P. Lopresti, H.F. Korth, J. Heflin, How carefully designed open resource sharing can help and expand document analysis research, in *Document Recognition and Retrieval XVIII, Part of the IS&T-SPIE Electronic Imaging Symposium* (2011), p. 78740O
9. B. Lamiroy, DAE-NG: A shareable and open document image annotation data framework, in *1st International Workshop on Open Services and Tools for Document Analysis, 14th IAPR International Conference on Document Analysis and Recognition* (2017), pp. 31–34
10. K.C. Santosh, B. Lamiroy, L. Wendling, Symbol recognition using spatial relations. Pattern Recognit. Lett. **33**(3), 331–341 (2012)
11. K.C. Santosh, L. Wendling, B. Lamiroy, Bor: Bag-of-relations for symbol retrieval. Int. J. Pattern Recognit. Artif. Intell. **28**(06), 1450017 (2014)
12. K. C. Santosh, B. Lamiroy, J.-P. Ropers, Inductive logic programming for symbol recognition, in *Proceedings of International Conference on Document Analysis and Recognition* (IEEE Computer Society, Washington, 2009), pp. 1330–1334
13. W. Min, Z. Tang, L. Tang, Recognition of dimensions in engineering drawings based on arrowhead-match, in *Proceedings of 2nd International Conference on Document Analysis and Recognition, Tsukuba (Japan)* (1993), pp. 373–376
14. G. Priestnall, R.E. Marston, D.G. Elliman, Arrowhead recognition during automated data capture. Pattern Recognit. Lett. **17**(3), 277–286 (1996)
15. L. Wendling, S. Tabbone, A new way to detect arrows in line drawings. IEEE Trans. Pattern Anal. Mach. Intell. **26**(7), 935–941 (2004)
16. K.C. Santosh, L. Wendling, S. Antani, G. Thoma, Scalable arrow detection in biomedical images, in *Proceedings of the IAPR International Conference on Pattern Recognition* (IEEE Computer Society, Washington, 2014), pp. 1051–4651
17. K.C. Santosh, N. Alam, P.P. Roy, L. Wendling, S.K. Antani, G.R. Thoma, A simple and efficient arrowhead detection technique in biomedical images. IJPRAI **30**(5), 1–16 (2016)
18. K.C. Santosh, P.P. Roy, *Arrow detection in biomedical images using sequential classifier* (Int. J. Mach. Learn, Cybern, 2017)
19. K.C. Santosh, L. Wendling, S. Antani, G.R. Thoma, Overlaid arrow detection for labeling regions of interest in biomedical images. IEEE Intell. Syst. **31**(3), 66–75 (2016)
20. F.T. Zohora, K.C. Santosh, Circular foreign object detection in chest x-ray images, in Recent Trends in Image Processing and Pattern Recognition, Revised Selected Papers, ed. by K.C. Santosh, M. Hangarge, V. Bevilacqua, A. Negi. Communications in Computer and Information. Science **709**, 391–401 (2017)
21. F.T. Zohora, K.C. Santosh, Foreign circular element detection in chest x-rays for effective automated pulmonary abnormality screening. Int. J. Comput. Vis. Image Process. **7**(2), 36–49 (2017)

22. F.T. Zohora, S.K. Antani, K.C. Santosh, Circle-like foreign element detection in chest x-rays using normalized cross-correlation and unsupervised clustering, in *Medical Imaging: Image Processing, Houston, Texas, United States, 10-15 February 2018* (2018), p. 105741V
23. K.C. Santosh, S. Antani, *Automated chest x-ray screening: can lung region symmetry help detect pulmonary abnormalities?* (IEEE Trans. Med, Imaging, 2017)

Index

K. C. Santosh, *Document Image Analysis*,
https://doi.org/10.1007/978-981-13-2339-3