Symbol Recognition using Spatial Relations

Santosh K.C., Bart Lamiroy, Laurent Wendling

▶ **To cite this version:**

Santosh K.C., Bart Lamiroy, Laurent Wendling. Symbol Recognition using Spatial Relations. Pattern Recognition Letters, 2012, 33 (3), pp.331-341. 10.1016/j.patrec.2011.09.040 . inria-00626996

Symbol Recognition using Spatial Relations

Santosh K.C.[a,*], Bart Lamiroy[b], Laurent Wendling[c]

[a]*LORIA – INRIA, 615 rue Jardin Botanique, 54600 Villers-lès-Nancy, France*
[b]*LORIA – Nancy Université, BP 239 - 54506 Vandoeuvre-lès-Nancy Cedex, France*
[c]*LIPADE, Université Paris Descartes, 75270 Paris Cedex 06, France*

Abstract

In this paper, we present a method for symbol recognition based on the spatio–structural description of a 'vocabulary' of extracted visual elementary parts. It is applied to symbols in electrical wiring diagrams. The method consists of first identifying vocabulary elements into different groups based on their types (e.g., *circle*, *corner*). We then compute spatial relations between the possible pairs of labelled vocabulary types which are further used as a basis for building an Attributed Relational Graph that fully describes the symbol. These spatial relations integrate both topology and directional information.

The experiments reported in this paper show that this approach, used for recognition, significantly outperforms both structural and signal–based state–of–the–art methods.

Keywords: Vocabulary, Spatial Relations, Attributed Relational Graph, Symbol Recognition.

1. Introduction

1.1. Motivation

Symbol recognition – the core part of graphical document image analysis and recognition systems – plays an important role in a variety of applications such as automatic recognition and interpretation of circuit diagrams [Okazaki

---

*Corresponding author
Email addresses:* `Santosh.KC@inria.fr` (Santosh K.C.), `Bart.Lamiroy@loria.fr` (Bart Lamiroy), `Laurent.Wendling@parisdescartes.fr` (Laurent Wendling)

et al., 1988], engineering drawings [Yang et al., 2007] and architectural drawings [Lladós et al., 2001; Valveny and Martí, 2003], maps [Samet and Soffer, 1996], musical notations [Rebelo et al., 2010], mathematical expressions [Chaudhuri and Garain, 2000], as well as optical characters [Yuen et al., 1998]. Therefore, a symbol can be defined as a graphical entity with a particular meaning in the context of a specific domain.

Research on graphics recognition has an extremely rich state–of–the–art literature, aimed to localise/recognise symbols depending on the applications. [Cordella and Vento, 2000; Lladós et al., 2002] show that these methods are particularly suited for isolated line symbols, not for composed symbols connected to a complex environment. In order to exploit the information embedded in those documents, one needs to be able to extract visual parts and formalise the possible links that exist between them. This combination of symbol localisation based on extracted visual parts is going to be the core of this paper and is very much inspired by a real world industrial problem [Tombre and Lamiroy, 2008; K.C. et al., 2009]. It consists in identifying a set of known symbols in aircraft electrical wiring diagrams, in order to bootstrap simulation algorithms. The main challenges come from the fact that the test symbols come in a wide variety of different forms. Symbols may either be very similar in shape, and only differ by slight details – or either be completely different from a visual point of view. Symbols may also be composed of other known and significant symbols and need not necessary be connected.

The rest of the paper is organised as follows. An overview of pertinent literature is given in Section 1.2, followed by a brief explanation of our proposed method in Section 2. We explain the way we describe symbols in Section 3, which mainly includes the concept of using spatial relations. We derive a symbol matching method from it in Section 4. Full experiments are reported in Section 5 and confront our method with current state–of–the–art algorithms. It includes a comprehensive experimental result analysis. We conclude in Section 6.

1.2. State–of–the–Art

1.2.1. Symbol Representations

Symbol recognition is a particular application of pattern recognition. Existing approaches, specifically those based on feature based matching, can be sorted into three classes: statistical, structural and hybrid. As respective

examples , among others, one can cite [Yang, 2005; Zhang et al., 2006; Lladós et al., 2001; Yang, 2005].

Under statistical approaches, global signal–based descriptors [Yuen et al., 1998; Kim and Kim, 2000; Tabbone et al., 2006; Belongie et al., 2002; Zhang and Lu, 2002, 2004] are usually quite fault tolerant to image distortions, since they tend to filter out small detail changes. This is unfortunately an inconvenient in our context. Moreover, they difficultly accommodate with connected or composite symbols. For instance, when symbols are combined, approaches that rely on centroid detection like [Yuen et al., 1998] tend to fail. Others, like Shape Context [Belongie et al., 2002] are sensible to occlusions on the symbol boundaries. Overall, they are generally not well adapted for capturing small detail changes, since they are specifically conceived to filter those out. In these statistical approaches, signatures are simple with low computational cost. However, discrimination power and robustness strongly depend on the selection of optimal set of features for each specific application.

Besides global signal–based descriptors, another idea is to decompose the symbols into either vector based primitives like points, lines, arcs *etc.* or into meaningful parts like *circles*, *triangles*, *rectangles etc.* These methods fall under structural approaches. They are then represented as Attributed Relational Graphs (ARG) [Bunke and Messmer, 1995; Conte et al., 2004], Region Adjacency Graphs (RAG) [Lladós et al., 2001], constraint networks [Ah-Soon and Tombre, 2001] as well as deformable templates [Valveny and Martí, 2003]. Their common drawback comes from error–prone raster–to–vector conversion. Those errors can increase confusions among different symbols. Furthermore, variability of the size of graphs leads to computational complexity in matching. However, structural approaches provide a powerful representation, conveying how parts are connected to each other, while also preserving generality and extensibility.

To describe the symbols, it is necessary to handle relations between the decomposed parts. The following paragraph gives an overview of existing work on spatial relations and their proper usages.

1.2.2. Spatial Relations

Effects of spatial relations on recognition performance have been examined comprehensively for scene understanding, document analysis and recognition tasks [Biederman, 1972; Bar and Ullman, 1993; Xiaogang et al., 2004; Pham and Smeulders, 2006]. Spatial relations can be either topological [Egenhofer and Franzosa, 1991; Egenhofer and Herring, 1991; Papadias

et al., 1995] directional [Bloch, 1999; Matsakis and Wendling, 1999; Wang and Keller, 1999] and metric in nature. For example, topological configurations are handled in [Xiaogang et al., 2004] with a few predicates like *intersection*, *interconnection*, *tangency*, *parallelism* and *concentricity* expressed with standard topological relations as described in [Egenhofer and Herring, 1991].

In a similar way, various directional relation models have been developed for a wide range of different situations.

- If the objects are far enough from each other, their relations can be approximated by their centres based on the discretised angle [Miyajima and Ralescu, 1994]. This approach is robust to small variations of shape and size.

- If they are neither too far nor too close, relations can be approximated by their *Minimum Bounding Rectangle* (MBR) [Lee and Hsu, 1992; E.Jungert, 1993; Papadias et al., 1995; Papadias and Theodoridis, 1997] as long as they are regular.

- Approaches like *Angle Histograms* [Wang and Keller, 1999] tend to be more capable of dealing with overlapping, something the previous ones have difficulties with. However, since they consider all pixels of a shape, their computational cost increases dramatically.

- Other methods, like *F–Histograms* [Matsakis and Wendling, 1999] use pairs of longitudinal sections instead of pairs of points, also at the cost of high time complexity.

- Another well–known approach uses fuzzy landscapes [Bloch, 1999], and is based on fuzzy morphological operators.

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. They rather have various degrees of sensitivity to or awareness of topological relations. While methods like [Xiaogang et al., 2004] focus on topological information only, our approach unifies both topological and directional information into one descriptor [K.C. et al., 2010] without any additional running time cost.

Placing spatial relations in the context of recognition and symbol description, 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

4

formalised in a mathematical way (*e.g.*, the 512 relations of the 9–intersection model [Egenhofer and Herring, 1991]). Therefore, qualitative and quantitative relations are another way to do categorisation of spatial relations. Consider an example, 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 upto a reasonable change of the objects' shape and position. Taking this into account, our work uses more natural relations than the all–or–none nature of standard relations [Freeman, 1975].

In the following section, we explain our proposed method by focusing on using spatial relations for describing and matching symbols.

2. Proposed Recognition Method

Our recognition method is based on a spatio–structural description of extracted visual parts that compose a symbol. This means that, to describe a symbol, we compute spatial relations between previously extracted visual parts. Without any other consideration, it is obvious that the size of the resulting relational graph is potentially very large and variable from one symbol to another. However, when grouping visual parts together according to their types (e.g., *circle*, *corner*) and by labelling them accordingly (see Section 3.1), we can eliminate all the combinatorial problems inherent to graph matching, without sacrificing recognition quality or expressive power.

We compute the spatial relations (see Section 3.2) between the distinct labelled attributes for building an Attributed Relational Graph (ARG – see Section 3.3), achieving at the same time integration of both topological and directional information.

Since each vertex represents a different class of visual parts, the graph has a uniquely and distinctly labelled vertex set. Vertex and edge matching thus becomes trivial and can be done in near–constant time.

3. Symbol Description

As mentioned in Section 2, we first define our visual vocabulary in Section 3.1. In Section 3.2 we explain the way we compute pairwise spatial relations and finally use both in Section 3.3 to build an ARG and completely describe the symbol.

3.1. Visual Vocabulary

We define a set of well controlled visual elementary parts as a *vocabulary* [K.C. et al., 2009]. While, in the general case, this vocabulary can be of any kind from any type of bag–of–features, related to what is visually pertinent in the application context under consideration, our current vocabulary is related to electrical symbols. It can be easily extended to adapt to other domains. Such visual elementary parts are extracted with the help of image treatment analysis operations as described in [Rendek et al., 2004]. Shortly, we discuss on how we accomplished it.

- *thick primitive:* We employ straight forward thin/thick separation by counting all *thick* connected components within the image. It simply uses standard skeletonisation using chamfer distance and computes the histogram of line thickness. An optimal cut value is computed from the histogram to distinguish between thick zones and thin zones.

- *circle primitive:* We use the algorithm as described in [Lamiroy and Guebbas, 2010] which is based on Random Sample Consensus minimization.

- *corner primitive:* We mainly consider four types of corners such as *North–East*, *North–West*, *South–East* and *South–West*. It uses simple template matching process *i.e.*, if the ratio of black and white pixels is greater than or equal to the template threshold, then the presence of corner is accessed.

- *extremity primitive:* We approach to detect loose end coordinates p_x from a given skeleton pixel where there is only a unique neighbouring pixel p_{c_1} connecting to the main skeleton, which itself is connected by a unique neighbouring pixel p_{c_2}.

Fig. 1 shows an illustration of visual primitives, extracted from two different symbols. Rather than using every detected element as a basis for expressing and computing spatial relations, we group them together by type as shown in Fig. 1. We denote the set of these generated groups as, $\sum_{\mathbb{T}} = \{\mathbb{T}_{thick}, \mathbb{T}_{circle}, \mathbb{T}_{corner}, \mathbb{T}_{extremity}\}$.
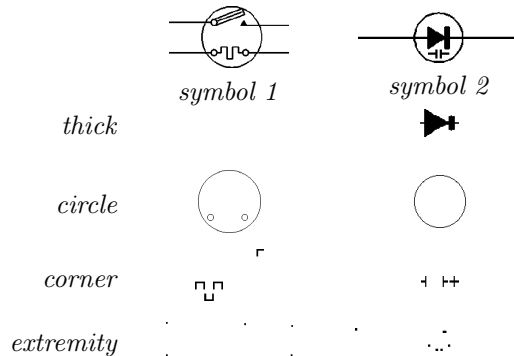
Figure 1: Illustration of vocabulary type.

3.2. Spatial Relation

In order to express the spatial distribution of the previously formed groups, we compute a spatial signature $\Re$ (defined further in Eq. (1)), expressing the spatial relations between two sets of pixels $\mathbb{A}$ and $\mathbb{B}$. This section explains in detail how it is computed.

Pairwise spatial relations are often expressed by using one of the objects as reference. For example, $\mathbb{A}$ is to the *right* of $\mathbb{B}$: *right*$(\mathbb{A}, \mathbb{B})$, where $\mathbb{B}$ is referenced. In our context, since the number of vocabulary types is not always the same for all symbols, it is difficult to take a particular type as a reference. To avoid such a difficulty, we first set up a unique reference point from each pair as shown in Step 1, hereafter. Then, we compute directional relations with respect to the reference point, thus avoiding potential ambiguity.

Step 1. Unique Reference Point Set

We consider a unique reference set $\mathbb{R}$, defined by the topology of the minimum bounding rectangles (MBR) of $\mathbb{A}$ and $\mathbb{B}$ and with the help of the 9-intersection model [Egenhofer and Herring, 1991]. In connection with [Renz and Nebel, 1998], $\mathbb{R}$ is either the common portion of two neighbouring sides in the case of *disconnected* MBRs or the intersection in the case of *overlapping*, *equal* or otherwise *connected* MBRs. To do this, we simply check topological relations between them in a $9-$dimensional binary space via the use of intersections of the boundaries, interiors and exteriors of two sets $\mathbb{A}$ and $\mathbb{B}$.

Depending on the obtained topological configurations, $\mathbb{R}$ can range from a point to a rectangular $2D$ area. In what follows, we define its

7

centroid point $\mathbb{R}_p$ as our reference point for computing spatial relation $\Re$ between $\mathbb{A}$ and $\mathbb{B}$.

Step 2. Directional Relation

For a given reference point $\mathbb{R}_p$, we cover the surrounding space at regular radial intervals of $\Theta = 2\pi/m$. As shown in Fig. 2 (a), a radial–line rotates over a cycle, and when intersecting with object $\mathbb{X}$ ($\mathbb{A}$ or $\mathbb{B}$), generates a boolean histogram $\mathcal{H}$,

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

where

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

This boolean histogram expresses whether there are any black pixels in direction θ_i. We extend this direction histogram, without loss of generality, to a histogram covering sectors defined by two successive angle values. Furthermore, rather than using boolean values, we can account for the percentage of pixels of the whole object lying in the general direction θ_i. Fig. 2 (b) gives an example for both types of histogram, boolean and percentage.

Applying this to both objects $\mathbb{A}$ and $\mathbb{B}$, our spatial relational signature $\Re(\mathbb{X}, \mathbb{R}_p)$ is the set of both histograms

$$\Re(\mathbb{X}, \mathbb{R}_p) = \{\mathcal{H}(\mathbb{A}, \mathbb{R}_p), \mathcal{H}(\mathbb{B}, \mathbb{R}_p)\}. \tag{1}$$

It is important to understand that we know the visual vocabulary types to which $\mathbb{A}$ and $\mathbb{B}$ belong (*cf.* Section 3.1). Defining a fixed arbitrary order on the set of types $\sum_{\mathbb{T}}$ solves the potential ordering problem when comparing two relational signatures.

Fig. 3 and Fig. 4 provide illustrations on hand–drawn and real–world examples respectively. The illustrations show how the reference point set $\mathbb{R}$ is obtained and show the corresponding histogram $\mathcal{H}$. They are analysed in the following section.
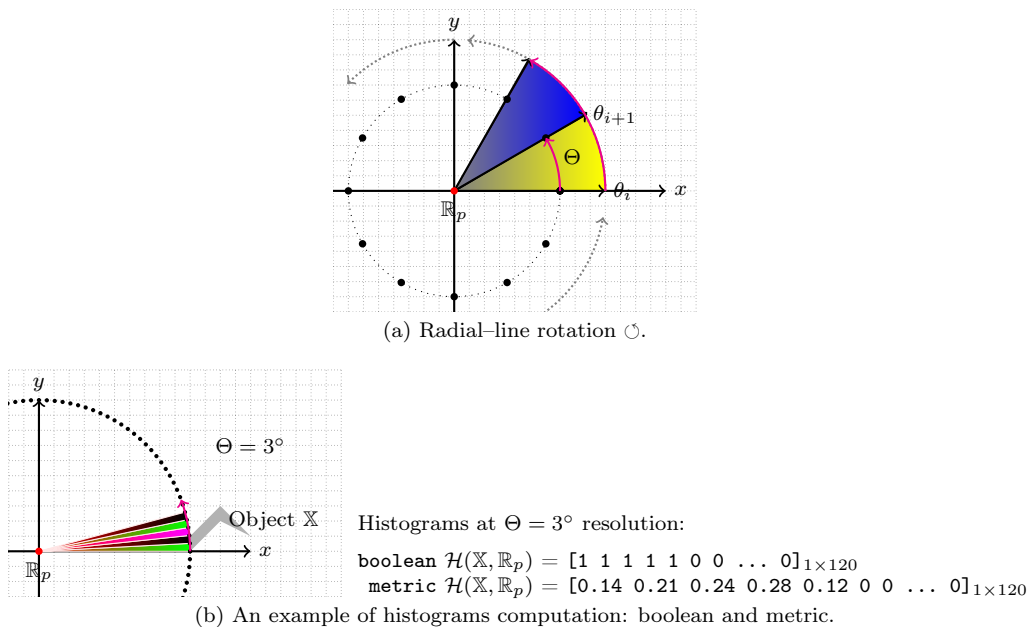
(a) Radial–line rotation ↺.

$\Theta = 3°$

Object $\mathbb{X}$

Histograms at $\Theta = 3°$ resolution:

boolean $\mathcal{H}(\mathbb{X}, \mathbb{R}_p) = [1\ 1\ 1\ 1\ 1\ 0\ 0\ \dots\ 0]_{1\times120}$
metric $\mathcal{H}(\mathbb{X}, \mathbb{R}_p) = [0.14\ 0.21\ 0.24\ 0.28\ 0.12\ 0\ 0\ \dots\ 0]_{1\times120}$

(b) An example of histograms computation: boolean and metric.

Figure 2: Computing spatial relations using radial–line rotation.

Illustrations. In Fig. 3 we show how our method adapts to different topological configurations. This section is not intended to give a full and formal evaluation of our approach, but rather to provide the user with an intuitive feeling on how it behaves. The figure shows various computed histograms in different configurations. These configurations were chosen to cover most topological relations between two objects one may encounter.

Let us consider the first three instances (a), (b) and (c). Keeping (a) as a reference image, we have changed a stroke thickness without changing relative positioning in (b) and moved objects closer while keeping identical topological configuration in (c). We observe that histograms do not show any significant difference. Scaling does not affect our method since $\mathcal{H}$ is normalised. In addition, the line rotation does not consider distance (*far* or *near*) information as long as it does not change the angular positioning.

For false *overlapping* configurations, as shown in (d), the coverage angle of $\mathcal{H}$ changes due to the change in structure (elongating horizontal limb in both objects).

For all *inclusion* configurations (like the false inclusion depicted in (f), but equally for full inclusion situations), our method does not produce any histogram for the component $\mathbb{X}$ which is either *contained* in or *covered* by
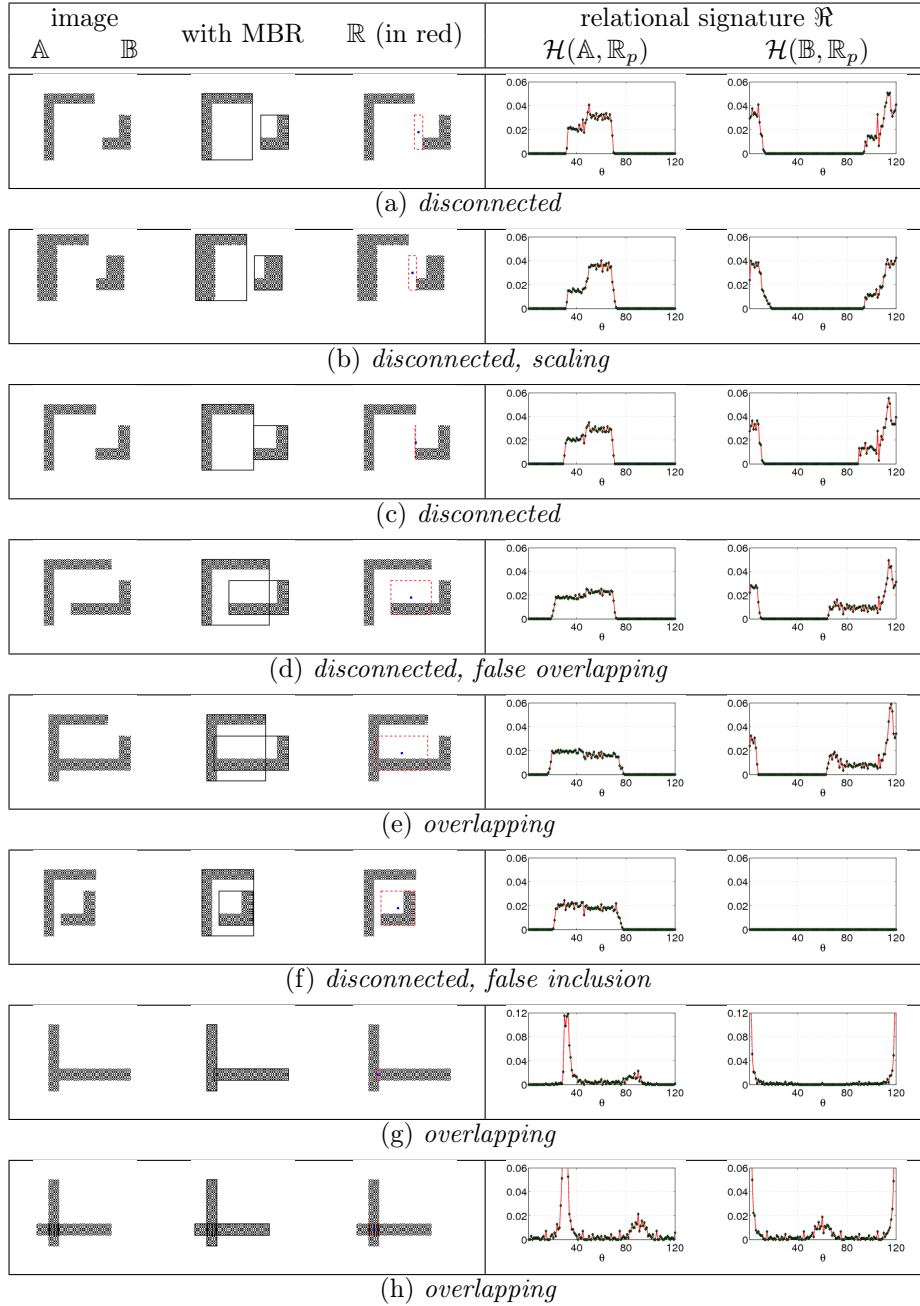
9

| image | with MBR | $\mathbb{R}$ (in red) | relational signature $\Re$ | |
| $\mathbb{A}$ $\mathbb{B}$ | | | $\mathcal{H}(\mathbb{A}, \mathbb{R}_p)$ | $\mathcal{H}(\mathbb{B}, \mathbb{R}_p)$ |

(a) *disconnected*

(b) *disconnected, scaling*

(c) *disconnected*

(d) *disconnected, false overlapping*

(e) *overlapping*

(f) *disconnected, false inclusion*

(g) *overlapping*

(h) *overlapping*

Figure 3: Histograms at $3°$ resolution for a few hand-drawn spatial objects $\mathbb{A}$ and $\mathbb{B}$, having different topological configurations.

10

image	with MBR	$\mathbb{R}$ (in red)	relational signature $\Re$

(a) *circle* and *corner*

$\mathcal{H}(circle, \mathbb{R}_p)$ $\mathcal{H}(corner, \mathbb{R}_p)$

(b) *circle* and *extremity*

$\mathcal{H}(circle, \mathbb{R}_p)$ $\mathcal{H}(extremity, \mathbb{R}_p)$

(c) *corner* and *extremity*

$\mathcal{H}(corner, \mathbb{R}_p)$ $\mathcal{H}(extremity, \mathbb{R}_p)$
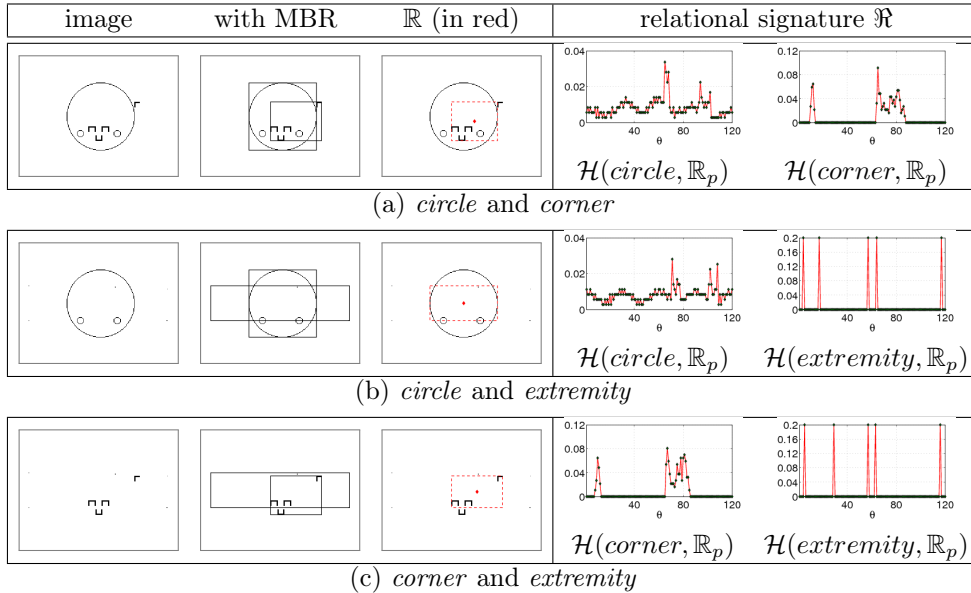
Figure 4: Histograms at 3° resolution for all possible pairs of vocabulary types from a *symbol 1* as shown in Fig. 1.

the other: it is simply $\mathcal{H}(\mathbb{X}, \mathbb{R}_p) = \emptyset$.

Besides, the difference of histograms between two *overlapping* cases in (g) and (h) can be observed in the middle of $\mathcal{H}(\mathbb{B}, \mathbb{R}_p)$ (between $40° - 80°$). This provides the fact that the method is able to discriminate slight changes in the object configurations even when identical topology exists.

Fig. 4 represents similar configurations taken from a real-world example, using the vocabulary extraction described in Section 3.1

Remarks. Our method captures the spatial information by the angular positions in the histogram. The magnitude of the histogram contains the structural information. Furthermore, running time does not depend on the size of the spatial objects as in the *Angle Histogram* approaches [Miyajima and Ralescu, 1994], for instance. Our method simply counts the number of pixels in every sector made by two consecutive radial–lines while rotating. However, running time is fixed and entirely depends on the parameter Θ (rotation step) that defines the resolution of $\mathcal{H}$ and the global size of the image. Its value is a trade-off between precision and execution time. We establish the optimal resolution for our application in Section 5.

3.3. Attributed Relational Graph

The vocabulary developed in Section 3.1 consists of a set of fixed label attributes, while the spatial relations between the attributes are the histograms described in Section 3.2. This gives us all the elements to express symbols as a complete ARG in which each vertex represents a distinct attribute type and the edges are labelled with a numerical expression of the spatial relations $\Re$.

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;

$F_A : V \to A_V$ is a function assigning labelled attributes to the vertices where A_V is the set of attributes type set $\sum_{\mathbb{T}}$ (*cf.* Section 3.1) and

$F_E : E \to \Re_E$ is a function assigning labels to the edges where $\Re$ represents the spatial relation of the edge E (*cf.* Section 3.2). Note that $\Re$ does not provide symmetry, $\Re(\mathbb{A}, \mathbb{B}) \neq \Re(\mathbb{B}, \mathbb{A})$. But, this can be solved by fixed ordering of V and $\mathbb{R}$ is not affected.

For instance, using *symbol 1* in Fig. 1 as an example, and its corresponding spatial relations in Fig. 4 we obtain the following ARG representation: $G = \{$

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

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

4. Symbol Recognition

Now that we have set up our ARG for symbol representation, we can define our recognition process. Recognition based on maximal similarity, measured by a matching score. The score is purely based on matching the corresponding relational signatures between the two given ARGs.

12

We then further extend the recognition by ranking database symbols based on the order of similarity, both of which will be explained in this section.

4.1. Matching

Following the ARG description in Section 3.3, let us consider two graphs:

$$
\begin{aligned}
G^q &= (V^q, E^q, F^q_A, F^q_E) \text{ for the query symbol and} \\
G^d &= (V^d, E^d, F^d_A, F^d_E) \text{ for the database symbol.}
\end{aligned}
$$

Let us remind that the set of vertices V, with $|V| = t$ and set of edges E, with $|E| = r$.

In order to explain our matching strategy, we are first taking the simplifying assumption that V^q and V^d are identical. In other words, both symbols contain items corresponding to identical vocabulary elements, but not necessarily sharing the same spatial arrangement. Since in our ARG every single vertex bears one distinct and unique attribute type, there is no cost in matching the vertices between G^q and G^d. As a consequence, matching edges is equally straightforward.

Since we have temporarily taken the assumption that V^q and V^d contain the same vocabulary elements, we can set up a bijective matching functions $\varphi : V^q \to V^d$ and $\sigma : E^q \to E^d$. 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. Also we consider that ordering is preserved over the vertices sets V^q and V^d. *I.e.* $v_1 < v_2 \Rightarrow \varphi(v_1) < \varphi(v_2)$.

Thanks to our fixed labelling of attribute types, corresponding $\Re$ 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,

$$
dist._{align}(G^q, G^d) = \sum_{r \in E} \delta \left(F^q_E(r), F^d_E(\sigma(r)) \right)
$$

where $\delta(a, b) = \sqrt{\sum_{l=1}^{L}(a_l - b_l)^2}$. This is actually a very simple and straightforward metric. Given the performances of our method reported in Section 5 there is no real need to have a more complex one, unless rotational invariance is needed.

Of course, the assumption that V^q and V^d share the exact same vocabulary is too strong. To generalise the previously described approach to any

situation, we define a binary (indicator) function $\tau_A^V : \Sigma_\mathbb{T} \to \{0,1\}$ to check the presence of vertices in the ARG, where the value of $\tau_A^V(\mathbb{T})$ is 1 if $\mathbb{T}$ is present in V and 0, otherwise. For example, for the *symbol 1* shown in Fig. 1, $\tau_A^V = [0,1,1,1]$. This refers to the absence of *thick* components and the presence of *circle*, *corner* and *extremity* components.

We can then use a simple edit cost between $\tau_A^{V^q}$ and $\tau_A^{V^d}$ defined by the number of edge deletions/insertions or substitutions. To do this, we first note the number of vertices to be deleted/inserted or substituted. Then consider the number of adjacent links.

$$dist._{edit}(G^q, G^d) = \sum_{p=1}^{P} c(o_p), \forall (o_1, \ldots, o_P) \in \Upsilon(G^q, G^d)$$

where $\Upsilon(G^q, G^d)$ denotes the set of edit paths transforming G^q into G^d and c, the edit cost function for operation o_p. Once virtual links exist (*i.e.*, null relation) after insertion for instance, edit cost is carried out as if matching has been done with relational alignments.

The final matching score or distance in the generic case $D(G^q, G^d)$ therefore is obtained from the fusion of edit cost and relational signatures alignment (reduced to the common node types between the two graphs). More formally, distance (matching score) between two given graphs is,

$$D(G^q, G^d) = \alpha \mathrm{dist.}_{align}(G^q, G^d) + (1-\alpha)\mathrm{dist.}_{edit}(G^q, G^d)$$

where $\alpha \in [0,1]$. The parameter α provides weight while matching. In our experiments we use $\alpha = 0.5$. The excellent results obtained and reported in Section 5.2 have not required us to tune this parameter further.

In the following, we give an example of how distance is computed.

Example. Consider $V^q = \{\mathbb{T}_1^q, \mathbb{T}_2^q\}$ in G^q and $V^d = \{\mathbb{T}_1^d, \mathbb{T}_2^d, \mathbb{T}_3^d\}$ in G^d. Then matching score between them is,

$$D(G^q, G^d) = \alpha \underbrace{\left[\frac{\delta(\Re_{\mathbb{T}_1^q,\mathbb{T}_2^q}, \Re_{\mathbb{T}_1^d,\mathbb{T}_2^d})}{}\right]}_{dist._{align}} + (1-\alpha) \underbrace{\begin{bmatrix} \delta(\Re_{\mathbb{T}_1^q,\mathbb{T}_3^q}, \Re_{\mathbb{T}_1^d,\mathbb{T}_3^d}) \\ + \\ \delta(\Re_{\mathbb{T}_2^q,\mathbb{T}_3^q}, \Re_{\mathbb{T}_2^d,\mathbb{T}_3^d}) \end{bmatrix}}_{dist._{edit}}$$

where $\Re_{x,y} = \Re(x,y)$. It is clear that $\mathbb{T}_3^q$ has to be inserted in G^q in order to transform it to G^d. As a consequence, virtual connections: $\Re_{\mathbb{T}_1^q,\mathbb{T}_3^q}$ and $\Re_{\mathbb{T}_2^q,\mathbb{T}_3^q}$

14

exits. Then matching is straightforward due to the labelled vertices in ARG. In addition, the weighting parameters are now useful to select either only $dist._{align}$ or taking both with equal as well as with different weights.

4.2. Ranking

The previously defined matching score conveys how similar/dissimilar a database symbol is with respect to a query. In order for similarity to be ranging from 1 to 0, we normalise $D(\textbf{.})$ to $[0, 1]$ by taking all database symbols: $\overline{D}(\textbf{.}) = \frac{D(\textbf{.}) - D^{min.}(\textbf{.})}{D^{max.}(\textbf{.}) - D^{min.}(\textbf{.})}$. Now, the similarity is,

$$Similarity(G^q, G^d) \quad = \quad 1 - \overline{D}(G^q, G^d).$$

Ranking can therefore be based on the decreasing order of similarity.

5. Experiments

In this section, we first give an overview of the symbols in our dataset and explain how we have labelled them with ground–truth. Then we discuss the evaluation metric, clarifying its proper usage for this application. Based on the metric, we perform a series of experiments and confront our method with the existing ones.

In the very beginning of the experiment, we consider the influence of different resolutions Θ in our relational signature. Once an optimal resolution is chosen, our spatial relation is compared with fundamental spatial relation models: *Cone–shaped* [Miyajima and Ralescu, 1994], *Angle Histogram* [Wang and Keller, 1999] and MBR [Papadias and Theodoridis, 1997]. Then we perform another assessment in order to make comparison of the complete method with the state–of–the–art approaches. For this, we first take a few representative global signal–based descriptors: region based *Zernike Moments* (ZM) [Kim and Kim, 2000], *Generic Fourier Descriptors* (GFD) [Zhang and Lu, 2002], *Shape Context* (SC) [Belongie et al., 2002] and $\mathcal{R}-signature$ [Tabbone et al., 2006], applied directly to the symbol. Then we take two recent pixel–based approaches: Statistical Integration of Histogram Array (SIHA) Yang [2005] and $2D$ kernel density Zhang et al. [2006] based symbol representation.

5.1. Dataset and Ground–truth Formation

Dataset. We work on a real world industrial problem to identify a set of different known symbols in aircraft electrical wiring diagrams as in [Tombre

and Lamiroy, 2008; K.C. et al., 2009]. Fig. 5 gives some examples of symbols in the database. Symbols may either be very similar in shape – and only differ by slight details – or either be completely different from a visual point of view. Symbols may also be composed of other known and significant symbols and need not necessary be connected. It is composed of roughly 500 different known symbols. Our dataset is completely unlabelled and imbalanced *i.e.*, neither ground–truth is given nor identical number of similar symbols exist for all queries.

Ground–truth Formation. Since there is no absolute ground–truth associated to our dataset, we have proceeded by using human validation, but by taking care of eliminating subjective bias. In order to achieve this we have asked 6 volunteers to manually select what they consider as "similar" symbols, for all queries executed in this section. Human evaluators have chosen the candidates which have similar visual overall appearance or which have significantly similar parts with respect to the chosen query. In our testing protocol, we consider that a result returned from an algorithm is correct if at least one human evaluator has selected the same result among the similar items. In more formal terms, for each query the "ground–truth" is considered to be the set of symbols formed by the union of all human selected sets. Fig. 5 provides a few examples. For instance, for query *a1*, evaluators have provided a list of symbols which they consider visually close, or containing parts that are visually close. The evaluators were not required to provide any ranking order nor degree of visual resemblance.

5.2. Experimental Protocol and Results

5.2.1. Experimental Protocol

Evaluation Metric. Our aim is not only limited to distinguish symbols but also extended to rank symbols in the provided lists. Ranking is related to similarity based on distance measure as described in Section 4.2. It becomes clear from Fig. 5 that there is a different number of pertinent documents in the database for each query. In order to be able to report retrieval results in a coherent way, we choose not to use classical precision and recall but to use the retrieval efficiency measure instead [Kankanhalli et al., 1995]. Retrieval efficiency has the advantage not to degenerate when the ranking parameter K grows bigger than the number of relevant items in the database, as would have been the case for precision and recall. For every chosen query, efficiency
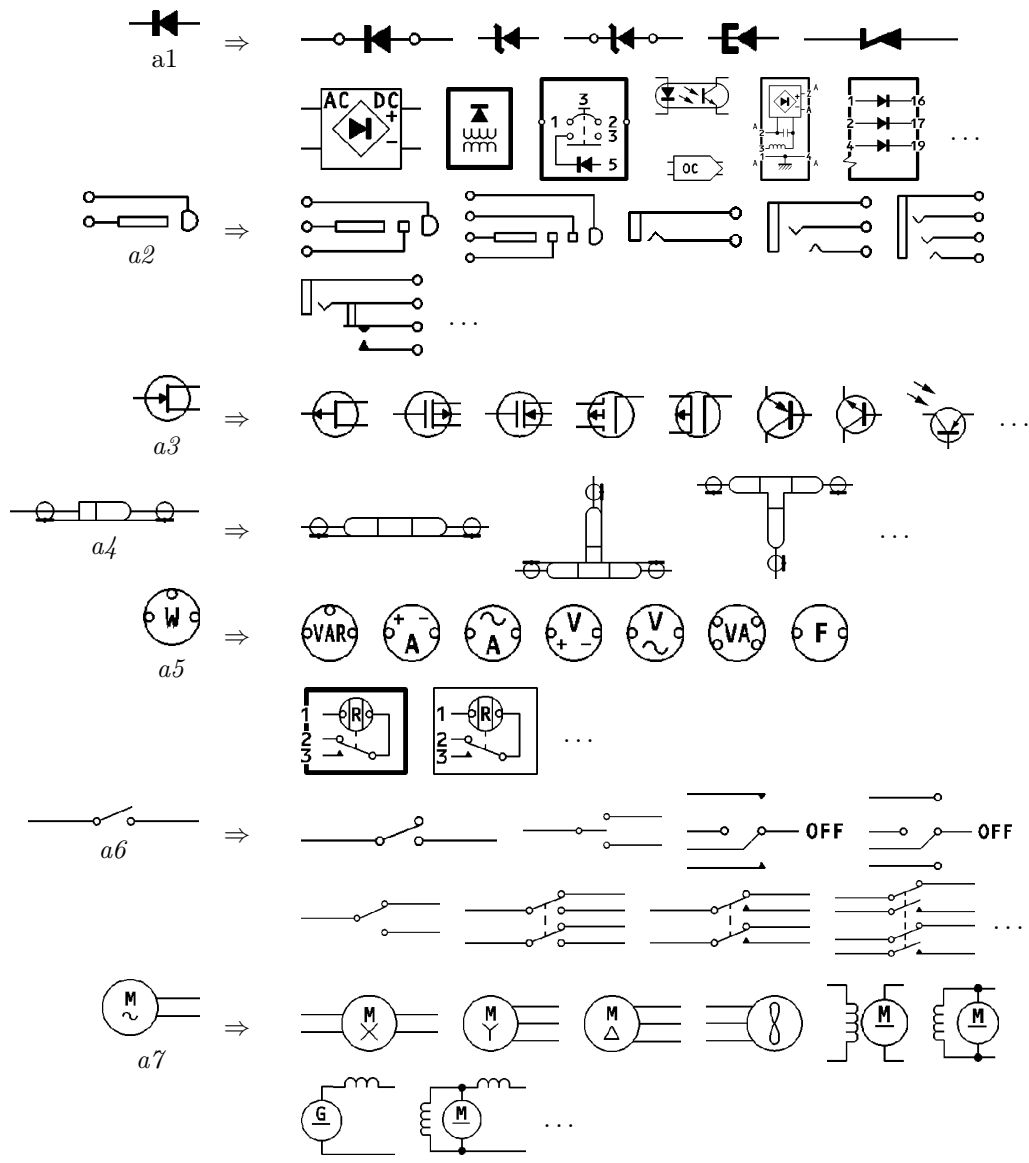
Figure 5: A sample of few electrical symbols. For every test symbol: *a1* to *a7*, a few relevant symbols are enlisted based on human evaluation. It consists of both linear as well as symbols in the composite form.

of retrieval for a given short list of size K is expressed as,

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

17

where n is the number of returned relevant symbols, N the total number of relevant symbols and K the number of ranked symbols requested. Note that η_K computes the traditional recall if $N \leq K$ and computes precision otherwise. The main advantage of this is that the average retrieval efficiency curve is not biased even with different quantities of ground–truth for different queries, while it happens for precision measures when $N < K$.

Matching Scope. Because of the fact we have a fixed set of labelled vertices (*i.e.*, *vocabulary* types) in our symbol description, we are able to control the matching scope for every chosen query by using a parameter s. Using the notation introduced in Section 4.1, we define s as $\delta(\tau_A^{V_q}, \tau_A^{V_d})$. Depending on the value of s different matching strategies can be applied:

$s \geq 0$: all candidates in the dataset are taken into consideration for matching.

$s \leq 1$: matching is only done between candidates differing by at most one vertex (*i.e.*, one vertex can be absent or supplementary).

$s = 0$: matching is done by candidates only having the exact same set of vertices (*i.e.*, $V^q = V^d$).

Therefore, we have applied the three different matching strategies to evaluate the behaviour of different methods with scopes ranging from $s \geq 0$ to $s = 0$. Our assumption is that candidates having same set of vertices as well as exact labels are similar either for their whole structure or part of it when in composite forms. This assumption has been experimentally validated.

5.2.2. Results

In this section we present a series of experiments establishing the performances of our approach. We address three specific issues:

1. What is the optimal set of parameters for our method?
2. How does our spatial relation model compare to other spatial relation models?
3. How well does our recognition model do with respect to state–of–the–art recognition models?

In all experiments, we have used *retrieval efficiency*, as described in Section 5.1. We compare the average retrieval efficiencies over the same 30 queries for all presented cases. These efficiency values have been computed for values of $K = 1$ to 10.

Resolution Parameter Determination. Our method, besides depending on the choice of the vocabulary, uses one main parameter: the resolution at which the angular histogram is computed[1]. Its value represents the trade-off between the optimal choice of resolution – and thus precision of spatio–structural information capture – and time/space requirements. Fig. 6(a) shows the result of a series of experiments with Θ varying over $\{1°, 3°, 5°, 9°\}$. For each of its values we have measured the retrieval efficiency on the same set of queries. Without surprise, the lower Θ, the better the results, independently of the matching strategies used.

Based on these results, and given the relatively low gain of efficiency between $3°$ and $1°$, we adopt the former for the rest of our experiments.

Other Spatial Relation Models. In order to compare our spatial relation model with others, we have adapted our ARG to function with those published in [Miyajima and Ralescu, 1994; Papadias and Theodoridis, 1997; Wang and Keller, 1999], and we have submitted them to the same testing protocol as described before. Fig. 6(b) shows their average retrieval efficiency. MBR outperforms all others in all situations. We shall further compare it to our method at the end of this section.

Global Signal–based Descriptors. In order to compare our method to other recognition methods, we have selected a set of major global signal–based shape descriptors described in Section 1.2 [Kim and Kim, 2000; Zhang and Lu, 2002; Belongie et al., 2002; Tabbone et al., 2006]. For GFD, we have tuned the parameters, and selected those values for radial and angular frequency that achieved the best recognition performance on our dataset: radial frequency 6 and angular 15. For *Shape Context*, only 70 sample points have been selected because of the presence of smaller size images in our database. In case of ZM, we have used 36 *zernike* functions of order less than or equal to 7. Also, we have taken radon image intensity over the projecting angle $[0, \pi[$ by default, for $\mathcal{R}-signature$. Unlike the methods based on spatial relations, we cannot establish different matching scopes, based on s as presented in Section 5.2.1 and used previously.

Again the same queries are presented and average retrieval efficiency is shown in Fig. 6(c). GFD seems to be performing the best among all tested

---

[1]The matching scope s, as introduced in Section 5.2.1 should not really be considered as a parameter, but as a measure of our method's robustness.

(a) Our method using different resolutions: $1°$, $3°$,$5°$ and $9°$.

(b) Fundamental spatial relation models.

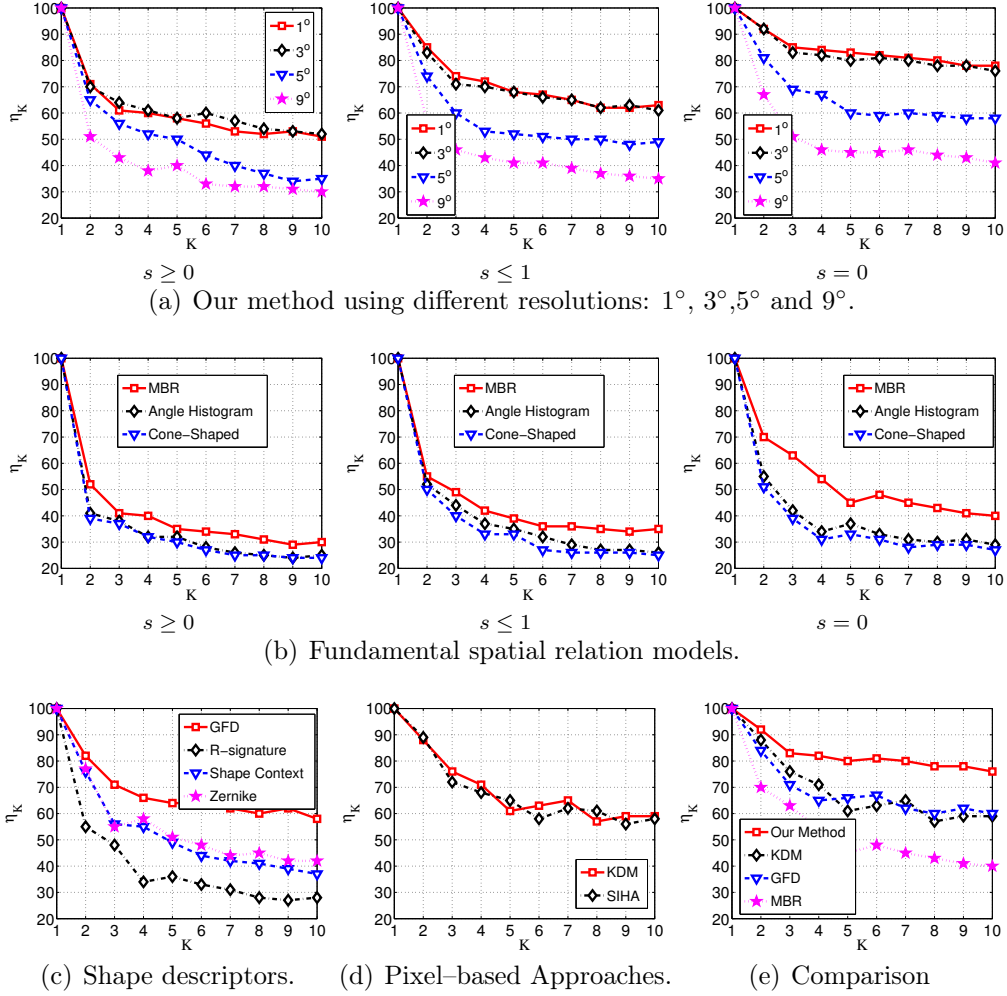(c) Shape descriptors.   (d) Pixel–based Approaches.   (e) Comparison

Figure 6: Average retrieval efficiency over requested list – 1 to 10: (a) Our method, (b) Fundamental spatial relation models, (c) Global signal–based descriptors, (d) Pixel–based approaches and (e) Comparison.

global signal–based descriptors in our setup.

Pixel–based Approaches. We have also compared our method with two pixel–based approaches specially designed for symbol recognition: Statistical Integration of Histogram Array (SIHA) [Yang, 2005] and Kernel Density Matching (KDM) [Zhang et al., 2006]. In SIHA, two different length–ratio and angle–ratio histograms are taken from every two pixels in reference to a

third pixel from the skeleton image. In KDM, skeleton symbols represent as $2D$ kernel densities and their similarity is measured by the *Kullback–Leibler divergence.* In Fig. 6(d), results are shown for both. In this test, we observe almost similar behaviour from the two. However, KDM performs slightly better, especially when also taking time complexity into account.

Overall, compared to our method, basic spatial relation models and global signal–based descriptors as well as recent pixel–based approaches have been lagging behind. Fig. 6(e) provides a comparison by taking the best of both classes: MBR from the spatial relation models, GFD from the global signal–based descriptors and KDM from the pixel–based approaches. Our method outperforms both with a significant difference in retrieval efficiency.

5.3. Discussions

In this section, the performance of the methods in response to the experimental results are analysed. Performance not only refers to retrieval efficiency but also to time complexity. In parallel, we discuss matching scope and its effect in ranking retrieved symbols.

To visually compare the results of our method with the best of breed solutions reported in Fig. 6(b), 6(c) and 6(d), we show a selection of queries in Fig. 7. They demonstrate the use of isolated as well as composed symbols as query. The first symbol on the top is always the chosen query and symbols are ranked from top to bottom (1 to 10) based on decreasing order of similarity. For query $Q1$, GFD and KDM come close to our method while MBR presents a notable difference. In case of query $Q2$, our method outperforms all others significantly. A similar situation happens for $Q3$.

Our method exploits spatio–structural description of the visual parts. The choice of the vocabulary types (*i.e.* collection of particular visual parts) is of course an important factor to its success. However, symbols like ⬛, ━⬛◄━ *etc.* are retrieved for the query ⬛◄ due to the presence of *thick* patterns. This shows that our relational signatures do not provide or use any shape information. Therefore, symbols having any *thick* pattern like, ◄,⬤, ⬤, ◢, ▼, ▼, ◣◪, ▲, ⬛, ▮, ◀ *etc.* are selected for ranking. However, spatial organisation of *thick* patterns with respect to other primitives helps to rank the best one first.

Running time has been measured in all experiments. An average running time (in sec.) for all methods is given below.

We used MATLAB 7.8.0 in Linux platform.

Our method has benefited from the way we describe the matching strategy (*cf.* Section 5.2.1). Symbol matching between the candidates which share the same sets of vertices with exact labels (*i.e.*, $s = 0$), is found to be the best among all. It sufficiently reduces time of matching to symbols which are obviously irrelevant. Similarly, this happens in those tests using basic spatial relations models. But for global signal–based descriptors as well as pixel–based approaches, running time increases with number of symbols in the dataset since matching scope does not exist.

6. Conclusions and Further Work

In this paper, we have presented a method to describe symbols using a specific Attributed Relational Graph via the use of spatial relations between the visual elementary parts. Each vertex represents all visual parts of a particular vocabulary type within the symbol. The edges represent the spatial relations between them. The proposed method is simple and flexible, and has the ability to express spatial relations between any number of visual parts. We have validated that such a description can be used for symbol recognition. Our method has proven to significantly outperform state–of–the–art basic spatial relation models, global signal–based descriptors and pixel–based approaches for symbol recognition.

22

Figure 7: Visual illustration of symbol retrieval and ranking for a few queries, showing the true (✓) and false (✗) retrieval.

Further work comprises the study of the influence of the weighting parameters in the matching score. Furthermore we are currently studying clustering techniques to enhance the discriminative power (and this enhance retrieval performance) of the *thick* component patterns. In addition, we are going to relate our work to [Bai et al., 2010] in order to see how both approaches can be combined to enhance overall performance.

References

Ah-Soon, C., Tombre, K., 2001. Architectural symbol recognition using a network of constraints. Pattern Recognition Letters 22 (2), 231–248.

Bai, X., Yang, X., Latecki, L., Liu, W., Tu, Z., 2010. Learning context-sensitive shape similarity by graph transduction. IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (5), 861 –874.

Bar, M., Ullman, S., 1993. Spatial context in recognition. Perception 25, 324–352.

Belongie, S., Malik, J., Puzicha, J., 2002. Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (4), 509–522.

Biederman, I., 1972. Perceiving real-world scenes. Science 177 (43), 77–80.

Bloch, I., 1999. Fuzzy relative position between objects in image processing: New definition and properties based on a morphological approach. Uncertainty Fuzziness and Knowledge-Based Systems 7 (2), 99–133.

Bunke, H., Messmer, B. T., 1995. Efficient attributed graph matching and its application to image analysis. In: Braccini, C., Floriani, L. D., Vernazza, G. (Eds.), Proceedings of International Conference on Image Analysis and Processing. Vol. 974 of Lecture Notes in Computer Science. Springer-Verlag, pp. 45–55.

Chaudhuri, B. B., Garain, U., 2000. An approach for recognition and interpretation of mathematical expressions in printed document. Pattern Analysis and Applications 3 (2), 120–131.

Conte, D., Foggia, P., Sansone, C., Vento, M., 2004. Thirty years of graph matching in pattern recognition. International Journal of Pattern Recognition and Artificial Intelligence 18 (3), 265–298.

Cordella, L. P., Vento, M., 2000. Symbol recognition in documents: a collection of techniques? International Journal on Document Analysis and Recognition 3 (2), 73–88.

Egenhofer, M., Franzosa, R., 1991. Point-set Topological Spatial Relations. International Journal of Geographical Information Systems 5 (2), 161–174.

Egenhofer, M., Herring, J. R., 1991. Categorizing Binary Topological Relations Between Regions, Lines, and Points in Geographic Databases. University of Maine, Research Report.

E.Jungert, 1993. Qualitative spatial reasoning for determination of object relations using symbolic interval projections. In: IEEE Symposium on Visual Languages. pp. 24–27.

Freeman, J., 1975. The modelling of spatial relations. Computer Graphics and Image Processing 4, 156–171.

Kankanhalli, M. S., Mehtre, B. M., Wu, J. K., 1995. Cluster-based color matching for image retrieval. Pattern Recognition 29, 701–708.

K.C., S., Lamiroy, B., Ropers, J.-P., 2009. Inductive logic programming for symbol recognition. In: Proceedings of International Conference on Document Analysis and Recognition. pp. 1330–1334.

K.C., S., Wendling, L., Lamiroy, B., 2010. Unified pairwise spatial relations: An application to graphical symbol retrieval. In: Ogier, J.-M., Liu, W., Lladós, J. (Eds.), Graphics Recognition. Achievements, Challenges, and Evolution. Vol. 6020 of Lecture Notes in Computer Science. Springer-Verlag, pp. 163–174.

Kim, W.-Y., Kim, Y.-S., 2000. A region-based shape descriptor using zernike moments. Signal Processing: Image Communication 16 (1-2), 95 – 102.

Lamiroy, B., Guebbas, Y., 2010. Robust and precise circular arc detection. In: Ogier, J.-M., Liu, W., Lladós, J. (Eds.), Graphics Recognition. Achievements, Challenges, and Evolution, 8th International Workshop, GREC

2009, La Rochelle, France, July 22-23, 2009. Selected Papers. Vol. 6020 of Lecture Notes in Computer Science. Springer-Verlag, pp. 49–60.

Lee, S.-H., Hsu, F.-J., 1992. Spatial Reasoning and Similarity Retrieval of Images Using 2D C-string Knowledge Representation. Pattern Recognition 25 (3), 305–318.

Lladós, J., Martí, E., Villanueva, J. J., 2001. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 23 (10), 1137–1143.

Lladós, J., Valveny, E., Sánchez, G., Martí, E., 2002. Symbol Recognition: Current Advances and Perspectives. In: Blostein, D., Kwon, Y.-B. (Eds.), GREC – Algorithms and Applications. Vol. 2390 of Lecture Notes in Computer Science. Springer-Verlag, pp. 104–127.

Matsakis, P., Wendling, L., 1999. A New Way to Represent the Relative Position Between Areal Objects. IEEE Transactions on Pattern Analysis and Machine Intelligence 21 (7), 634–643.

Miyajima, K., Ralescu, A., 1994. Spatial Organization in 2D Segmented Images: Representation and Recognition of Primitive Spatial Relations. Fuzzy Sets and Systems 2 (65), 225–236.

Okazaki, A., Tsunekawa, S., Kondo, T., Mori, K., Kawamoto, E., 1988. An automatic circuit diagram reader with loop-structure-based symbol recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence 10 (3), 331–341.

Papadias, D., Sellis, T., Theodoridis, Y., Egenhofer, M. J., 1995. Topological relations in the world of minimum bounding rectangles: a study with r-trees. SIGMOD Record 24 (2), 92–103.

Papadias, D., Theodoridis, Y., 1997. Spatial relations, minimum bounding rectangles, and spatial data structures. International Journal of Geographical Information Science 11 (2), 111–138.

Pham, T. V., Smeulders, A. W. M., 2006. Learning spatial relations in object recognition. Pattern Recognition Letters 27 (14), 1673–1684.

26

Rebelo, A., Capela, G., Cardoso, J. S., 2010. Optical recognition of music symbols: A comparative study. International Journal on Document Analysis and Recognition 13 (1), 19–31.

Rendek, J., Masini, G., Dosch, P., Tombre, K., 2004. The search for genericity in graphics recognition applications: Design issues of the qgar software system. In: Marinai, S., Dengel, A. (Eds.), Proceedings of International Workshop on Document Analysis Systems. Vol. 3163 of Lecture Notes in Computer Science. Springer-Verlag, pp. 366–377.

Renz, J., Nebel, B., 1998. Spatial reasoning with topological information. In: Spatial Cognition, An Interdisciplinary Approach to Representing and Processing Spatial Knowledge. Springer-Verlag, pp. 351–372.

Samet, H., Soffer, A., 1996. Marco: Map retrieval by content. IEEE Transactions on Pattern Analysis and Machine Intelligence 18, 783–798.

Tabbone, S., Wendling, L., Salmon, J.-P., 2006. A new shape descriptor defined on the radon transform. Computer Vision and Image Understanding 102 (1), 42–51.

Tombre, K., Lamiroy, B., 2008. Pattern recognition methods for querying and browsing technical documentation. In: Proceedings of Iberoamerican Congress on Pattern Recognition. Springer-Verlag, pp. 504–518.

Valveny, E., Martí, E., 2003. A model for image generation and symbol recognition through the deformation of lineal shapes. Pattern Recognition Letters 24 (15), 2857–2867.

Wang, X., Keller, J., 1999. Human-Based Spatial Relationship Generalization Through Neural/Fuzzy Approaches. Fuzzy Sets and Systems 101, 5–20.

Xiaogang, X., Zhengxing, S., Binbin, P., Xiangyu, J., Wenyin, L., 2004. An online composite graphics recognition approach based on matching of spatial relation graphs. International Journal on Document Analysis and Recognition 7 (1), 44–55.

Yang, R., Lu, T., Cai, S., 2007. A dynamic-rule-based framework of engineering drawing recognition and interpretation system. In: ICIC: International Conference on Advanced intelligent computing theories and applications. Springer-Verlag, pp. 1006–1017.

27

Yang, S., 2005. Symbol recognition via statistical integration of pixel-level constraint histograms: A new descriptor. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (2), 278–281.

Yuen, P. C., Feng, G.-C., Tang, Y. Y., 1998. Printed chinese character similarity measurement using ring projection and distance transform. International Journal of Pattern Recognition and Artificial Intelligence 12 (2), 209–221.

Zhang, D., Lu, G., 2002. Shape-based image retrieval using generic fourier descriptor. Signal Processing: Image Communication 17, 825–848.

Zhang, D., Lu, G., 2004. Review of shape representation and description techniques. Pattern Recognition 37 (1), 1–19.

Zhang, W., Wenyin, L., Zhang, K., 2006. Symbol recognition with kernel density matching. Pattern Recognition 28 (12), 2020–2024.