

Analyse des documents graphiques :  
une approche par reconstruction d'objets

15 juin 2007

# Chapitre 1

## Gestion des connaissances graphiques, état de l'art

### 1.1 Introduction

Au cours du chapitre précédent nous avons présenté différentes méthodes d'extraction de primitives graphiques. Nous avons vu dans quelle mesure ces méthodes étaient complémentaires. Nous avons également illustré les différents niveaux d'extraction selon lesquels se décomposent ces méthodes. Nous en avons conclu à la complémentarité des méthodes d'extraction et donc à l'intérêt de leur combinaison. Cette combinaison soulève cependant le problème de l'échange des primitives graphiques entre les méthodes. Plus largement, elle soulève le problème de la gestion des connaissances dites graphiques au sein des systèmes d'analyse des documents. Nous abordons cette problématique dans ce chapitre. Nous rappelons d'abord brièvement<sup>1</sup> différentes notions sur les systèmes à base de connaissances, et plus largement la gestion des connaissances, dans la section (1.2). Dans la section (1.3) nous dressons un état de l'art sur la gestion des connaissances dites graphiques et de son application aux systèmes d'analyse des documents. Finalement, dans la section (1.4) nous concluons.

### 1.2 Systèmes à base de connaissances

#### 1.2.1 Introduction

Les systèmes informatiques capables d'exploiter des connaissances sont qualifiés de Système à Base de Connaissances (SBC)<sup>2</sup> dans la littérature [Haton 91]. La recherche sur les SBC est très active, plusieurs ouvrages de synthèse y sont consacrés : [Ullman 89], [Haton 91], [Bachimont 94], [Ermine 96] et [Leondes 00]. Les premiers SBC sont apparus dans le courant des années 1970. Ces systèmes sont alors qualifiés de systèmes

---

<sup>1</sup>De par l'absence de consensus sur ce domaine cette introduction n'est évidemment pas exhaustive.

<sup>2</sup>Knowledge-Based System (KBS) [Leondes 00]

experts [Farreny 89]. La particularité de ces systèmes est de séparer les programmes des connaissances. Ils sont basés sur l'utilisation d'un ensemble de règles (les connaissances) exploitées par un moteur d'inférence (le programme). Dans le courant des années 1980, la limite de ces systèmes a conduit la communauté de l'intelligence artificielle à considérer d'autres approches comme le Raisonnement à Partir de Cas (RàPC) [Kolodner 93] ou les systèmes à base de contraintes [Hentenryck 89]. Le développement de ces nouvelles approches a élargi les systèmes experts aux SBC. Ceci rend aujourd'hui la définition d'un SBC très difficile, chaque approche étant en effet très spécifique. [Haton 91] en synthétise une architecture sur la figure (1.1). Il décompose les SBC en deux parties principales : une base de connaissances et un mécanisme d'exploitation de ces connaissances.

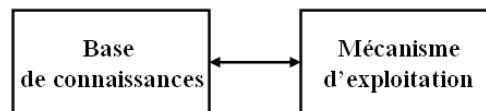


FIG. 1.1 – Système à Base de Connaissances (SBC) [Haton 91]

Les aspects connaissances des SBC ont fait émerger au début des années 1990 une nouvelle discipline de recherche qualifiée d'Ingénierie des Connaissances (IC)<sup>3</sup> ou de Gestion des Connaissances (GC)<sup>4,5</sup> [Charlet 02]. Cette dernière vise à la mise en oeuvre des concepts, méthodes et modèles dégagés par les sciences humaines<sup>6</sup> à des fins instrumentales [Otman 97]. Elle est très active, plusieurs ouvrages de synthèse y sont consacrés [Tuthill 90], [Kayser 97] et [Holsapple 04]. L'IC se situe donc à la frontière de l'informatique et des sciences humaines. Dans ce contexte, il est difficile de formaliser précisément et de façon exhaustive cette discipline. Dans la suite de cette section, nous définissons et illustrons différents concepts utilisés en IC.

## 1.2.2 Données, informations et connaissances

Le concept central de l'IC est celui de connaissances. Il est difficile d'en proposer une définition vu l'absence de consensus existant sur ce concept en IC [Charlet 02]. Malgré cette absence, on définit aujourd'hui d'un point de vue des techniques informatiques<sup>7,8</sup> [Kayser 97] le concept des connaissances selon une hiérarchie à trois niveaux : données, informations et connaissances. [Silvent 05] présente cette hiérarchie dans la définition (1). La figure (1.2) (a) l'illustre graphiquement.

<sup>3</sup> Knowledge Engineering [Tuthill 90]

<sup>4</sup> Knowledge Management [Holsapple 04]

<sup>5</sup> La GC est plutôt perçue comme la pratique de l'IC [Charlet 02].

<sup>6</sup> psychologie, neuro-sciences et linguistique [Gordon 97]

<sup>7</sup> Ce point de vue est celui adopté par la communauté anglophone [Holsapple 04].

<sup>8</sup> Nous reportons le lecteur à [Bachimont 04] pour une définition orientée sciences humaines.

**Définition 1 *Données-Informations-Connaissances*** : Les données sont le résultat d'observations ; les informations sont le résultat de l'interprétation de ces données ; les connaissances définissent la manière dont les données et les informations vont être exploitées.

Les données correspondent aux valeurs numériques classiquement manipulées par tout système informatique : entiers, caractères, flottants, . . . Lorsque ces données sont interprétées, on les qualifie alors d'informations. La figure (1.2) (b) illustre cette interprétation des données [Saidali 02]. On définit finalement les connaissances comme la manière dont les données et/ou informations sont exploitées au sein d'un système [Kayser 97]. Néanmoins on emploie également dans la littérature [Saidali 02], par abus de langage<sup>9</sup>, le terme connaissances du point de vue des connaissances implicites introduites par les concepteurs au sein des systèmes.

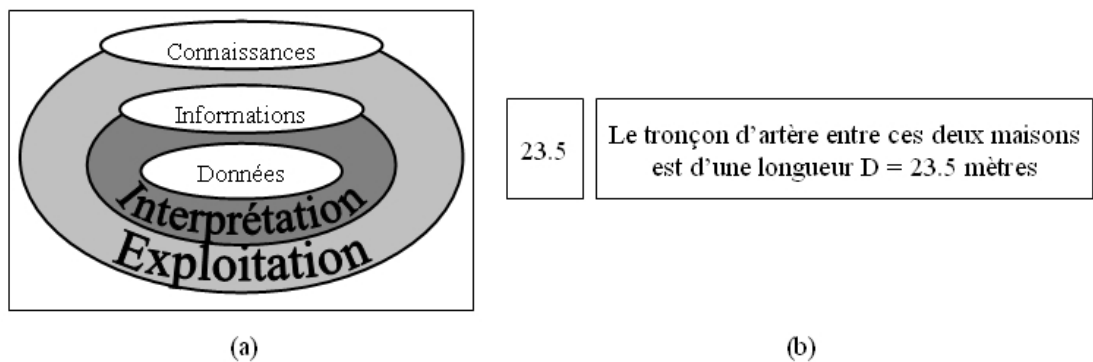


FIG. 1.2 – (a) données, informations et connaissances (b) donnée vs information

### 1.2.3 Représentation, formalisme et modèle

Les connaissances sont présentes au niveau procédural (interne) et déclaratif (externe) dans les systèmes [Haton 91]. Dans les deux cas, elles peuvent être transcrites de différentes façons : on parle de représentation des connaissances. La représentation des connaissances est d'abord fonction d'un formalisme. La notion de formalisme a été introduite dans les années 1950 par [Klages 50]. Celui-ci la définit comme la pensée par signes purs. Le but est alors d'atteindre les résultats de la pensée sans l'effort de la pensée. La notion de formalisme est donc large et dépasse le cadre de l'IC. Restreint à cette discipline [Valencia 00] en propose la définition suivante :

**Définition 2 *Formalisme*** : Série de symboles définis utilisée pour représenter les connaissances au sein des systèmes.

<sup>9</sup>Nous nous permettrons également cet abus dans ce manuscrit.

Il existe différents types de formalismes employés au sein des SBC [Kayser 97]. Certains formalismes simples s'apparentent plus à de la représentation des données (matrices, vecteurs, ...) [Lucas 86]. D'autres plus complexes s'apparentent à de la représentation des connaissances (réseaux sémantiques, schémas, ...) [Kayser 97]. Ils se déclinent en différentes familles, parmi elles citons principalement<sup>10</sup> les logiques de description et les formalismes à base de graphes [Kayser 97]. La figure (1.3) illustre la différence entre ces deux familles.

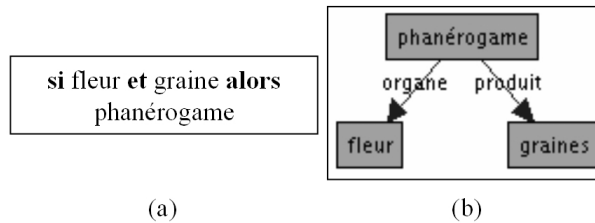


FIG. 1.3 – (a) logique (b) graphe

Il est possible d'exploiter un formalisme donné de différentes façons au sein des SBC : on utilise alors des modèles. [Seguela 01] définit ce concept par la définition (3). La figure (1.4) l'illustre avec deux modèles graphiques<sup>11</sup> différents (b) (c) basés sur le formalisme graphe d'un symbole (a).

**Définition 3 Modèle :** Nous appelons modèle une représentation des connaissances réalisée dans un formalisme donné et suivant un but donné.

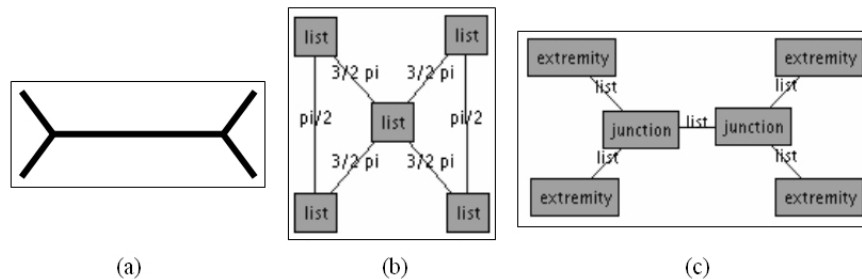


FIG. 1.4 – (a) symbole (b) modèle à base de liste (c) modèle à base de jonction

Ce concept de modèle, du fait de son utilisation dans de nombreuses disciplines, est difficile à fixer [Seguela 01]. En IC la frontière entre modèle et formalisme est floue, certains auteurs les emploient indifféremment pour désigner le même concept [Kayser 97] [Holsapple 04]. À travers la définition (3) et l'exemple de la figure (1.4), le concept de modèle est présenté dans ce manuscrit comme une instance de formalisme.

<sup>10</sup>Nous reportons le lecteur à [Kayser 97] pour une présentation détaillée des familles de formalisme.

<sup>11</sup>Nous reprenons plus en détail ces modèles page 21 de ce chapitre.

### 1.2.4 Langage, opérationnalisation et interopérabilité

Les connaissances représentées au niveau déclaratif (externe) dans les SBC sont regroupées au sein de bases de connaissances [Kayser 97]. Les connaissances représentées dans ces bases sont évidemment en adéquation avec les formalismes et modèles de représentation adoptés par les systèmes en interne. L'implémentation de ces bases repose sur l'utilisation de langages de représentation des connaissances [Barry 01]. [Otman 97] définit un langage de représentation des connaissances de la manière suivante :

**Définition 4** *Langage de représentation des connaissances : Langage qui permet de modéliser ou de représenter des connaissances.*

Il existe de nombreux langages de représentation des connaissances. Ces langages peuvent se classer en deux catégories selon qu'ils soient formels et/ou opérationnels. [Barry 01] définit la distinction entre ces catégories dans la définition (5).

**Définition 5** *Langage de représentation formel et opérationnel : Parmi les langages de représentation des connaissances, suivant que les spécifications obtenues sont ou non exécutables, nous distinguons les langages de représentation (simplement) formels et les langages de représentation opérationnels.*

Les langages formels se limitent à la représentation statique des connaissances. Dans le cas de ces langages, on entend alors par connaissances celles implicites introduites par les concepteurs/utilisateurs des systèmes. La figure donne deux exemples de langages formels appliqués à la représentation de graphe<sup>12</sup>. Les langages opérationnels définissent en plus de la représentation formel un ensemble d'axiomes spécifiant la façon dont les données peuvent être utilisées [Fürst 04]. Ces derniers sont plus puissants que les langages formels grâce à leur capacité à produire de nouvelles données [Barry 01]. Le langage logique PROLOG est un exemple type de langage opérationnel. Celui-ci est basé sur l'utilisation de règles, il a largement été utilisé dans les systèmes experts [Farreny 89]. La figure (1.6) suivante donne un exemple de règle PROLOG.

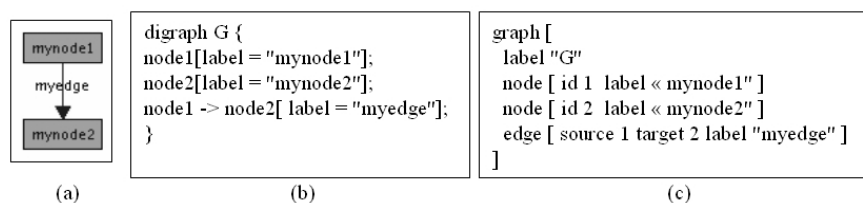


FIG. 1.5 – (a) graphe (b) GraphViz (c) GML

<sup>12</sup> extrait du site : [http://www.compendiumdev.co.uk/stareast2003/practical1\\_2.htm](http://www.compendiumdev.co.uk/stareast2003/practical1_2.htm)

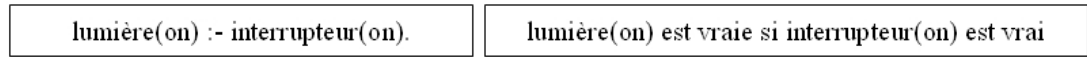


FIG. 1.6 – Exemple de règle PROLOG

L'IC vise à porter<sup>13</sup> les formalismes et modèles vers des langages de représentation des connaissances [Fürst 04]. Cette étape est du ressort de l'ingénieur des connaissances (ou cognicien) [Otman 97]. Dans le cas d'un langage formel le terme généralement employé pour cette étape est celui de transcription. Dans le cas d'un formalisme opérationnel on parle alors d'opérationnalisation<sup>14</sup>. [Fürst 04] en propose la définition (6) suivante. Néanmoins, ce terme opérationnalisation est également employé dans la littérature (par abus de langage)<sup>15</sup> pour toute transcription et/ou exploitation de langages de représentation des connaissances [Charlet 02]. Par analogie, on emploie également le terme **opérateurs** dans la littérature technique [Kernighan 99] pour différencier les méthodes et algorithmes (niveau concept) de leurs implémentations (niveau physique) au sein des systèmes.

**Définition 6 *Opérationnalisation*** : *Processus qui consiste à plonger les connaissances dans un langage opérationnel de représentation, en accord avec le type d'application visé.*

Une fois les formalismes, modèles, et langages définis [Defude 05] soulève la problématique de l'interopérabilité au sein ou entre SBC. L'interopérabilité est un concept bien connu en informatique [André 01]. C'est une propriété selon laquelle plusieurs systèmes, qu'ils soient identiques ou radicalement différents, peuvent communiquer sans ambiguïté. [Defude 05] distingue deux types d'interopérabilité : technique et sémantique. L'interopérabilité technique est du domaine des techniques informatiques (réseaux et formats). Elle concerne la mise en place de normes de communication ouvertes et communautaires [Seghrouchni 04]. Dans le cadre de l'IC, celle-ci se traduit par les capacités des SBC à comprendre et/ou à traduire différents langages de représentation des connaissances [Winter 02]. Cette dernière est évidemment possible si les SBC concernés emploient des formalismes et des modèles de représentation a priori similaires. Le cas contraire est la problématique de l'interopérabilité sémantique. Celle-ci est définie par [Defude 05] à travers la définition (7). Elle concerne les problèmes de manipulation et d'échange des connaissances entre différents systèmes (et/ou opérateurs) malgré les divergences de représentations.

**Définition 7 *Interopérabilité sémantique*** : *Capacité de manipuler de manière cohérente les informations provenant des différents systèmes. Par exemple, reconnaître que tel objet A dans tel système correspond à tel objet B dans un autre système.*

<sup>13</sup>Cette étape est précédée par celle de conceptualisation, nous reportons le lecteur à [Gruber 95].

<sup>14</sup>Deux orthographes existent dans la littérature (un *n* ou deux), nous avons adopté la plus courante.

<sup>15</sup>Nous nous permettrons également cet abus dans la suite du manuscrit.

### 1.2.5 Conclusion

Dans cette section nous avons présenté l'Ingénierie (ou Gestion) des Connaissances à travers ses concepts généraux. Il est difficile de formaliser cette discipline vu l'absence de consensus existant. Néanmoins la présentation des concepts de représentation, formalisme, modèle, langage de représentation, et opérationnalisation permet de l'appréhender. Nous avons également soulevé la problématique de interopérabilité, en particulier sémantique, entre systèmes. Celle-ci concerne la manipulation et l'échange des connaissances entre différents systèmes (et/ou opérateurs) malgré les divergences de représentations. À travers ces notions, on s'aperçoit que la définition d'un SBC est fonction du couple formalisme/opérationnalisation. On qualifie de SBC dans la littérature tout système exploitant un "degré suffisant" de formalisme, utilisant un langage de représentation des connaissances, et l'opérationnalisant.

## 1.3 Gestion des connaissances graphiques

### 1.3.1 Introduction

Les connaissances graphiques correspondent à la mise en oeuvre des données graphiques au sein des systèmes informatiques. Ces données sont composées des primitives graphiques, mais aussi<sup>16</sup> des structures, utilisées pour représenter les objets graphiques. Les structures sont basées sur des formalismes à base de graphes. En effet les graphes permettent une représentation spatiale des données ce qui se prête intuitivement aux primitives graphiques. La figure (1.7) suivante donne deux exemples de graphes de primitives graphiques. Le premier (a) est basé sur des primitives graphiques composantes connexes (fond et forme) extraites d'une image de document [Badawy 02]. Le second (b) est basé sur des primitives graphiques polygones construites à partir d'un ClipArt [Fonseca 04].

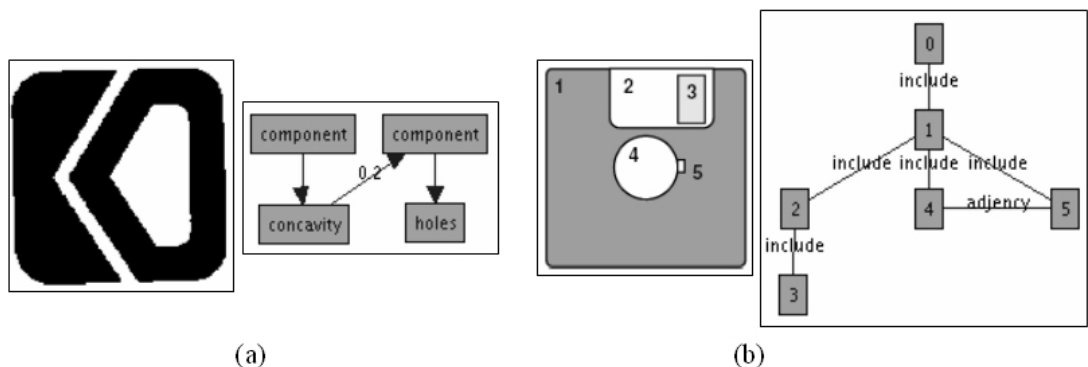


FIG. 1.7 – (a) graphe de composantes (b) graphe de polygones

<sup>16</sup>L'utilisation de ces structures n'est pas systématique pour toutes les représentations.



La notion de connaissances graphiques a été introduite dans le courant des années 1970 dans le cadre des Systèmes d'Information Géographique (SIG)<sup>17</sup> [Worboys 04]. Ces SIG ont pour but : "l'acquisition, la modélisation, la manipulation, la recherche, l'analyse, et la présentation des données géographiquement référencées"<sup>18</sup>. Un nombre important de travaux ont été publiés sur les langages SIG [Mark 89]. Ceux-ci ont pour but la représentation des données géographiques pour leur manipulation, visualisation, et recherche. L'application des connaissances graphiques dans les systèmes d'analyse des documents s'est développée (indépendamment des SIG) vers la fin des années 1980 [Okazaki 88]. La problématique en est différente, les langages employés ont pour but d'assister les systèmes d'analyse des documents dans la tâche de reconnaissance et d'interprétation. Parmi ces systèmes de nombreux ont été appliqués à l'interprétation d'images de document graphique : [Tombre 91], [Joseph 92], [Nardelli 93], [Collin 93], [Vaxiviere 94], [Pasternak 95], [Hartog 96], [Xu 97], [Ah-Soon 01], [Coüasnon 01], [Weindorf 02], [Yan 04], ... Dans le courant des années 90 la révolution du document électronique a fait émerger la problématique de l'indexation de document [Doermann 98]. Différents travaux ont alors concerné l'indexation de documents graphiques vectoriels ([Dunlop 95], [Love 01], [Smith 01], [Yaner 02], [Fonseca 04], et [Sciascio 04], ...) et images<sup>19</sup> ([Kasturi 88], [Seong 93] [Lorenz 95], [Mahmood 95], [Huang 97], [Tombre 03], [Barbu 04], ...). Ces travaux ont fait émerger les problématiques de méta-données dans les connaissances graphiques. Parallèlement différents travaux sur le web sémantique ont porté sur les problèmes de représentation des connaissances graphiques via les langages graphiques vectoriels web (SVG, VML, ...) [Gould 99] [Herman 02] [Chen 04].

Dans la suite de cette section nous présentons un état de l'art sur la gestion des connaissances graphiques. Celle-ci concerne aujourd'hui différents domaines et donc différentes problématiques : recherche, indexation, visualisation, ... Notre approche dans ce manuscrit vise à la combinaison des méthodes d'extraction de primitives graphiques pour l'analyse des documents. Cependant cette combinaison soulève la problématique de l'interopérabilité sémantique entre les opérateurs. Cette dernière concerne l'échange de connaissances graphiques représentées de différentes façons entre les opérateurs. Afin de répondre à cette problématique nous nous sommes intéressés dans notre état de l'art à la représentation des connaissances graphiques (formalismes et modèles). Dans la suite de cette section nous présentons tout d'abord les formalismes utilisés pour la représentation des connaissances graphiques : primitives graphiques et formalismes à base de graphes. Nous comparons par la suite différentes représentations des connaissances employées au sein des systèmes de la littérature. Basé sur les conclusions de cette comparaison nous introduisons alors la multi-représentation, et comment celle-ci permet de répondre à la problématique de l'interopérabilité sémantique entre opérateurs d'extraction. Enfin, nous concluons sur cet état de l'art.

---

<sup>17</sup>Geographic Information System (GIS)

<sup>18</sup>"capture, modeling, manipulation, retrieval, analysis, and presentation of geographically referenced data" [Worboys 04].

<sup>19</sup>Nous ignorons ici les systèmes d'indexation d'image couleur et reportons le lecteur à [Smeulders 00].

### 1.3.2 Les primitives graphiques

Les primitives graphiques sont les objets graphiques élémentaires manipulés dans les systèmes d'analyse des documents [Shih 89]. Elles servent de premier niveau de représentation intermédiaire entre les niveaux image et symbolique (figure (1.8)), ces derniers étant des niveaux d'abstraction (de l'acquisition à l'interprétation) pour la représentation idéale des objets [Bahr 02]. Les primitives graphiques sont ensuite structurées dans des objets graphiques de plus haut niveau représentant le niveau d'abstraction symbolique correspondant à l'image.

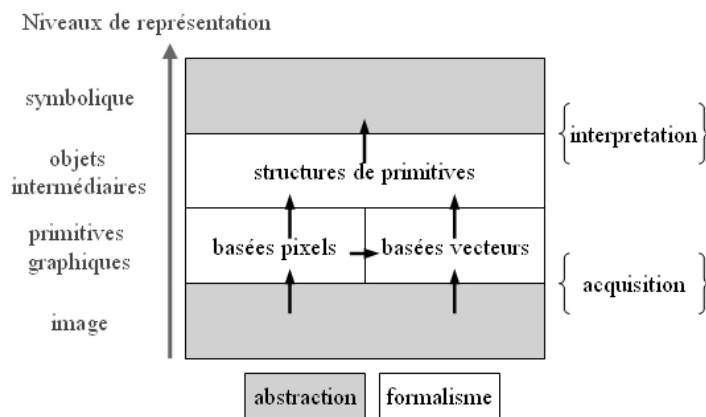


FIG. 1.8 – Niveaux de représentation

Il existe deux formalismes principaux utilisés pour représenter les primitives graphiques, à base de pixels (primitives : raster, composantes, plages, pixels, ...) et à base de vecteurs (primitives : vecteurs, arcs, courbes, ...) [Chhabra 98] [Cordella 00] [Lladós 02]. La figure (1.9) présente un schéma relationnel de ces formalismes ainsi que de leurs primitives associées.

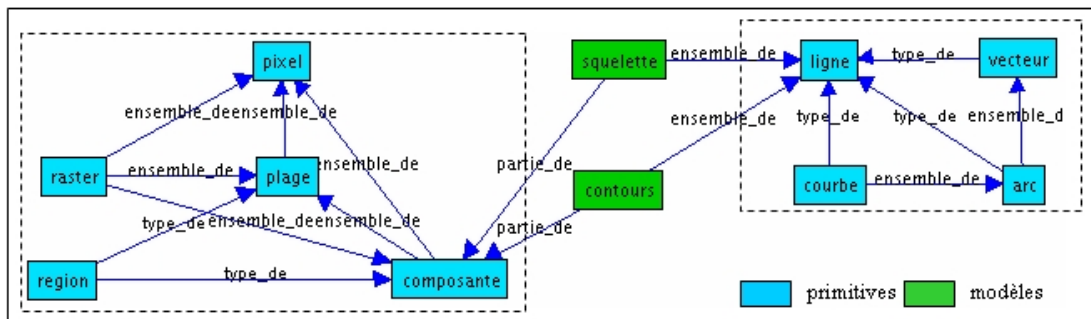


FIG. 1.9 – Schéma relationnel des formalismes et des primitives graphiques<sup>20</sup>

<sup>20</sup>Ce schéma ne représente que les relations principales existantes entre les primitives graphiques.

Le lien entre le formalisme à base de pixels et celui à base de vecteurs est assuré par les modèles contours et squelettes des composantes connexes. Ces derniers décrivent ainsi les composantes connexes de façon linéaire permettant l'extraction ultérieure de primitives graphiques à base de vecteurs. C'est une approche largement utilisée par les systèmes de vectorisation, de nombreux travaux y sont consacrés ([Fan 98], [Ramel 00], [Zou 00], [Tombre 00], [Mejbri 02], ...). Le formalisme à base de pixels permet une représentation brute (non vectorielle) de l'image. Il est plus adapté pour la description des formes pleines sur les images de document. Il est fréquemment utilisés dans les systèmes de reconnaissance de symboles [Lladós 02] et de caractères [Mori 92] caractérisés par ces formes pleines. Dans les deux formalismes les différentes primitives graphiques répondent à de fortes relations de composition (*ensemble\_de*, *partie\_de*) et de spécialisation (*type\_de*). De même, ces primitives graphiques peuvent être toutes structurées en objets graphiques de plus haut niveau. Ces objets représentent alors le niveau d'abstraction symbolique correspondant à l'image. Les structures utilisées sont basées sur des formalismes à base de graphes ; nous les présentons dans la section suivante.

### 1.3.3 Formalismes à base de graphes

Les graphes sont un outil mathématique fondamental<sup>21</sup> en informatique pour la modélisation. La figure (1.10) en présente deux exemples dit non orienté (a) et orienté (b). La théorie des graphes date du début des années 60 [Berge 83]. Depuis lors les graphes ont été utilisés dans divers domaines d'application (automatisme, vision, web, ...) [Gross 03].

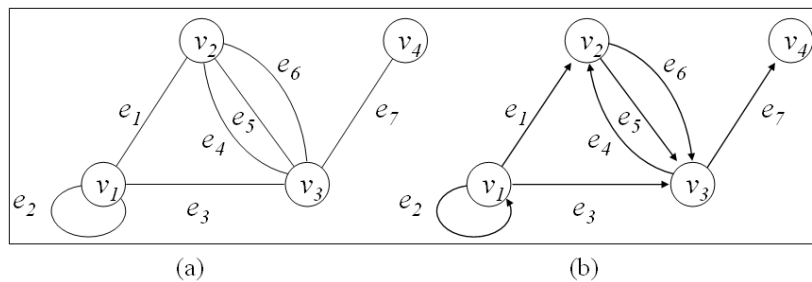


FIG. 1.10 – Exemple de graphe (a) non orienté (b) orienté

L'application des graphes à la représentation des connaissances a été introduite par [Quillian 68]. Celui-ci a développé le concept de réseaux sémantiques qui constitue le premier formalisme à base de graphes pour la représentation des connaissances [Sowa 99]. La figure (1.11) (a) en donne un exemple. Ces réseaux permettent d'exprimer des relations entre objets hiérarchisés. Les noeuds y représentent les divers objets et les arcs des propriétés, des relations, ou des actions possibles sur ces objets.

<sup>21</sup>Nous reportons le lecteur à l'Annexe A pour une introduction sur les graphes.

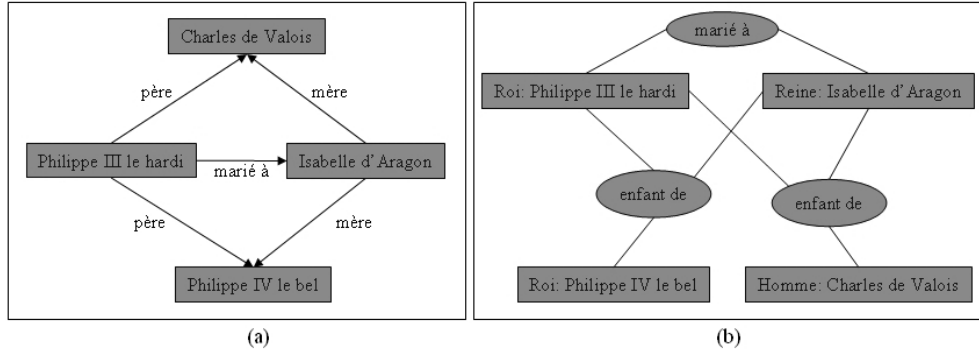


FIG. 1.11 – graphe<sup>22</sup>(a) sémantique (b) conceptuel

[Sowa 76] a étendu par la suite les graphes sémantiques aux graphes conceptuels. La figure (1.11) (b) en donne un exemple. La particularité des graphes conceptuels est la dissociation des notions de concepts (Philippe III le hardi, Isabelle d' Aragon) et de relations (marié à, enfant de). Ces graphes conceptuels sont dits bipartis, c'est à dire utilisant deux classes de noeuds. Cette structure bipartie permet une séparation explicite des différentes connaissances (concepts et relations). Les graphes conceptuels sont largement utilisés en ingénierie des connaissances pour la gestion des ontologies<sup>23</sup> [Gandon 02].

Parallèlement à l'émergence des graphes sémantiques [Minsky 75] a proposé le formalisme des schémas<sup>24</sup>. Les schémas correspondent à des descriptions d'entités conceptuelles représentant des objets réels "The Watergate Hotel" ou abstraits "Cola Wars" [Kuipers 94]. Ils sont essentiellement définis par leurs relations avec les autres schémas représentées à l'aide d'attributs<sup>25</sup>. La figure (1.12) donne un exemple de schéma (b) pour la description d'un arbre généalogique (a) [Kuipers 94]. Le schéma décrivant *Adam* (b) est composé de trois attributs {*sex*, *spouse*, *child*}. Des relations d'héritage peuvent être également exprimées entre schémas, celles-ci ont pour intérêt essentiel l'héritage des attributs [Lassila 90]. Les schémas ont été appliqués principalement aux systèmes experts [Farreny 89].

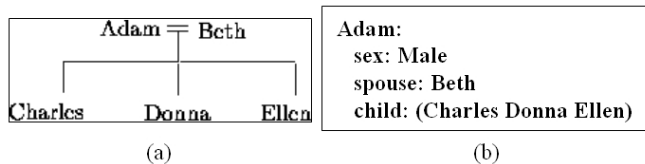


FIG. 1.12 – (a) arbre généalogique (b) schéma

<sup>22</sup> Exemples extraits de : <http://www.lirmm.fr/%7Ebaget/downloads/baget-01-pres.ppt>

<sup>23</sup> Nous reportons le lecteur à [Gandon 02] sur cet aspect.

<sup>24</sup> frames

<sup>25</sup> slots

Dans le courant des années 70 un autre formalisme a émergé [Euzenat 98], celui des langages Orienté(s)-Objet (O.O). Ces langages sont basés sur le concept d'objet introduit par [Nygaard 63] dans le langage de programmation Simula. Ils sont aujourd'hui largement utilisés<sup>26</sup> en informatique pour le développement d'application et reposent sur des mécanismes désormais bien connus (héritage, polymorphisme, ...) [Spaccapietra 05]. L'application des langages O.O à la représentation des connaissances s'est faite à travers les langages dits "post-frame" [Euzenat 98]. Ceux-ci ont enrichi les langages de représentation à base de schémas avec les concepts issus des langages de programmation O.O. Dans ces langages le schéma étendu à l'objet correspond à l'unité logico-sémantique de la connaissance [Spaccapietra 05]. Cette unicité est d'ailleurs à l'opposé des autres formalismes qui cherchent généralement à séparer les connaissances des mécanismes de raisonnement [Gensel 92]. Ces langages ont également adopté une distinction générique-spécifique des schémas en classes et instances [Valtchev 99]. Parmi ces langages citons les deux précurseurs : KRL<sup>27</sup> [Bobrow 77] et FRL<sup>28</sup> [Roberts 77]. Malgré les parallèles évidents existants entre les langages O.O de programmation et de représentation des connaissances ceux-ci poursuivent des buts différents [Valtchev 99]. Les premiers servent à l'écriture des programmes tandis que les seconds sont destinées à la modélisation. Les langages O.O de représentation des connaissances ont été utilisés dans diverses applications [Pachet 97] [Euzenat 99] comme les bases de données et les ontologies.

Les formalismes présentés précédemment (réseaux sémantiques, graphes conceptuels, schémas et langages O.O) s'appliquent à la représentation des connaissances d'un point de vue général. Ils sont majoritairement utilisés dans les problèmes d'ingénierie des connaissances et d'intelligence artificielle [Barr 89]. Dans les années 70 [Barrow 71] a introduit une spécialisation du graphe particulièrement utilisé en reconnaissance des formes [Milgram 93] qualifié de graphe relationnel attribué<sup>29</sup>. Celui-ci est présenté au travers de la définition (8). Dans ce graphe différents attributs sont utilisés pour labelliser les noeuds et les arcs. Les attributs employés peuvent être très divers et sont fonction du domaine d'application comme la vision (région) [Ahmadyfard 00] ou le génie logiciel (index textuel) [Sartipi 01]. Cette diversité dans l'attribution fait que le graphe relationnel attribué peut être considéré comme une généralisation des réseaux sémantiques [Barthélemy 05]. De même, elle rend le graphe relationnel attribué particulièrement adapté à la représentation des connaissances graphiques. Les noeuds et les arcs des graphes décrivent alors les structures des objets graphiques et les attributs les différentes primitives. La figure (1.13) donne un exemple extrait des travaux de [Seong 93]. De nombreux travaux ont été consacrés aux problèmes de l'appariement et de la modélisation des primitives graphiques par des graphes relationnels attribués : [Tsai 79], [Sanfeliu 83], [Seong 93], [Foggia 99], [Zhang 04], ...

---

<sup>26</sup> Nous reportons le lecteur à l'Annexe B pour une introduction sur les langages O.O.

<sup>27</sup> Knowledge Representation Language

<sup>28</sup> Frame Representation Language

<sup>29</sup> Attributed Relational Graph (ARG)

**Définition 8 Graphe relationnel attribué :** Soit  $G=(V,E,A,\mu,\nu)$

$V = \{v_1, v_2, \dots, v_n\}$  est un ensemble de noeuds

$E = \{e_1, e_2, \dots, e_m\}$  est un ensemble d'arcs

$A = \{a_1, a_2, \dots, a_u\}$  est un ensemble fini d'attributs

$\mu : v_i \mapsto a_q$  une fonction d'attribution des noeuds

$\nu : e_j \mapsto a_r$  une fonction d'attribution des arcs

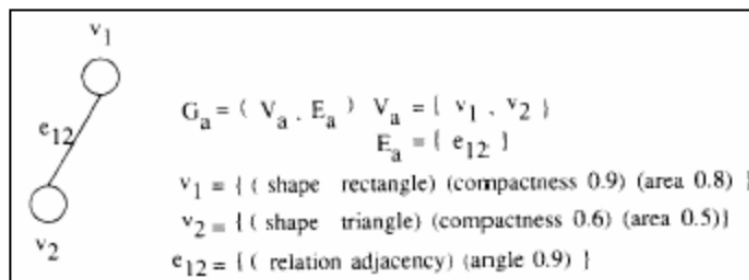


FIG. 1.13 – Exemple de graphe relationnel attribué [Seong 93]

Nous avons présenté ici divers formalismes à base de graphes exploitables pour la représentation des connaissances graphiques : réseaux sémantiques, graphes conceptuels, schémas, langages O.O et graphes relationnels attribué. Ces divers formalismes permettent de représenter des structures de type graphe ainsi que divers attributs. L'utilisation d'un formalisme donné sera fonction des besoins d'un SBC en matière de représentation et d'opérationnalisation des connaissances. Dans la suite de cette section nous présentons l'exploitation de ces formalismes pour la représentation des connaissances graphiques.

### 1.3.4 Représentations des connaissances graphiques

Dans cette section nous avons présenté les formalismes utilisés pour la représentation des connaissances graphiques : primitives graphiques et formalismes à base de graphes. Nous présentons maintenant une comparaison des représentations des connaissances graphiques (formalismes et modèles) employées dans la littérature. Dans ce manuscrit nous avons montré qu'il existait un grand nombre de systèmes manipulant des données graphiques. Pour notre comparaison nous n'avons tout d'abord considéré que les systèmes séparant strictement<sup>30</sup> leurs connaissances graphiques des autres catégories<sup>31</sup> de connaissances. Ensuite, parmi le grand nombre de systèmes répondant à cette contrainte nous avons sélectionné ceux qui nous ont semblé les plus représentatifs.

<sup>30</sup>Ceci exclue une majorité des systèmes SIG.

<sup>31</sup>Comme les connaissances du domaine (ou métier) [Pasternak 95].

Ces systèmes et représentations des connaissances graphiques associées sont présentés dans les tableaux (1.1) et (1.2). Les représentations sont comparées selon trois entrées : primitives, formalismes, et modèles. Nous avons également indiqué dans ces tableaux les différentes applications des systèmes. L'entrée primitives renseigne sur les différentes primitives graphiques extraites et manipulées par les systèmes. Nous n'avons dans cette comparaison considéré que les primitives bas niveau (raster, vecteur, ...). L'entrée formalismes renseigne elle sur les formalismes employés pour structurer les primitives graphiques. Ces formalismes sont de types formels mais aussi opérationnels<sup>32</sup>. Enfin l'entrée modèles renseigne sur la nature des modèles de représentation utilisés dans chacun des systèmes. Nous avons regroupé ces modèles en trois classes principales : région, contour, et squelette. Dans la suite de cette section nous comparons les différentes représentations utilisées au sein des systèmes selon ces trois entrées.

En premier lieu, les systèmes manipulent de nombreuses primitives graphiques communes. Les types de primitive manipulés sont fonction des algorithmes d'extraction employés au sein des systèmes. De nombreux systèmes ([Joseph 92], [Han 94], [Yu 97], [Lladós 01], [Hilaire 04], ...) emploient une approche classique de vectorisation (squelette et/ou contours) en deux étapes (chaînage, polygonalisation). Ils ont donc recours à des primitives de type liste de points et vecteur. Certains systèmes ([Kiyko 95], [Popel 02]) n'emploient pas de polygonalisation, ils exploitent uniquement des primitives graphiques de type liste de points. D'autres systèmes ([Wenjin 99], [Coüasnon 01], [Song 02]) extraient directement des primitives graphiques de type vecteur et arc en une étape. Différents systèmes ([Burge 95], [Yu 97], [Burge 98], [Coüasnon 01], [Badawy 02], [Leplumey 02]) utilisent des primitives graphiques "non linéaires" comme les composantes connexes et les plages. D'autres systèmes ([Pasternak 93], [Weindorf 02], [Fonseca 04]) exploitent directement des primitives graphiques de type vecteur et arc en entrée à partir de documents graphiques vectoriels (DXF, ClipArt, ...). Enfin la majorité des systèmes n'emploie qu'un type donné de primitives graphiques, cependant certains systèmes en emploient plusieurs simultanément ([Yu 97], [Coüasnon 01]).

Les formalismes de représentation employés dans les systèmes sont avant tout formels. La majorité des systèmes ([Seong 93], [Messmer 95], [Huang 97], [Ahmed 00], [Badawy 02], [Yan 04], ...) se réclament d'une approche type reconnaissance des formes. Ils emploient alors des formalismes formels de type structure relationnelle attribuée. Certains systèmes ([Huang 97], [Yu 97], [Ahmed 00]) regroupent les primitives graphiques en de simples ensembles sous forme de liste relationnelle attribuée. Les listes sont alors exploitées par des méthodes d'appariement de primitives graphiques (vecteurs et images [Yu 97], chaînes [Huang 97], ...). La majorité des systèmes ([Seong 93], [Messmer 95], [Burge 98], [Lladós 01], [Yan 04], ...) exploitent des structures plus complexes sous forme de graphe relationnel attribué. Ces systèmes utilisent alors des méthodes d'appariement de graphes exploitant ([Seong 93], [Lladós 01]) ou non ([Messmer 95], [Fonseca 04]) les primitives graphiques durant l'appariement.

---

<sup>32</sup>Nous n'aborderons ici que les aspects formels de ces formalismes opérationnels.

Références	Primitives	Formalismes	Modèles	Applications
[Joseph 92]	raster, liste de points, vecteur	schémas	squelette	interprétation de dessins techniques
[Pasternak 93]	vecteur, arc	spécification de contraintes	squelette	interprétation de dessins techniques
[Seong 93]	vecteur	graphe relationnel attribué	région	reconnaissance de symboles
[Wu 93]	raster, liste de points, vecteur	orienté-objet	squelette	interprétation de dessins techniques
[Han 94]	raster, liste de points, vecteur	graphe relationnel attribué	contour	interprétation de dessins techniques
[Burge 95]	raster, composante	graphe relationnel attribué	région	segmentation texte/graphique
[Messmer 95]	raster, liste de points, vecteur	graphe relationnel attribué	squelette	reconnaissance de symboles
[Kiyko 95]	raster, liste de points	grammaire	squelette	interprétation de dessins techniques
[Huang 97]	raster, liste de points, vecteur	liste relationnelle attribué	région	reconnaissance de logos
[Yu 97]	raster, liste de points, vecteur, composante	liste relationnelle attribué	squelette	interprétation de dessins techniques
[Burge 98]	raster, liste de plages	graphe relationnel attribué	squelette	interprétation de dessins techniques
[Lee 98]	raster, liste de points, vecteur	graphe relationnel attribué	squelette	interprétation de plans cadastraux
[Ahmed 00]	raster, liste de points, vecteur	liste relationnelle attribué	squelette	reconnaissance de symboles

TAB. 1.1 – Représentations des connaissances graphiques (1)



Références	Primitives	Formalismes	Modèles	Applications
[Ramel 00]	raster, liste de points, vecteur	graphe relationnel attribué	contour	interprétation de plans cinématiques
[Ah-Soon 01]	raster, liste de points, vecteur, arc	spécification de contraintes	squelette	interprétation de dessins techniques
[Coüasnon 01]	raster, vecteur, composante	grammaire	région	reconnaissance de formulaires, partitions, formules
[Lladós 01]	raster, liste de points, vecteur	graphe relationnel attribué	région	reconnaissance de symboles
[Badawy 02]	raster, composante	graphe relationnel attribué	région	reconnaissance de symboles
[Popel 02]	raster, liste de points	graphe relationnel attribué	squelette	compression d'images
[Song 02]	raster, vecteur, arc	orienté-objet	squelette	interprétation de dessins techniques
[Leplumey 02]	raster, composante	graphe relationnel attribué	région	reconnaissance de formules mathématiques
[Weindorf 02]	vecteur, arc	grammaire	région	interprétation de documents graphiques vectoriels
[Barbu 04]	raster, liste de points	graphe relationnel attribué	région	reconnaissance de symboles
[Hilaire 04]	raster, liste de points, vecteur, arc	graphe relationnel attribué	squelette	interprétation de dessins techniques
[Fonseca 04]	vecteur	graphe relationnel attribué	région	indexation de graphiques vectoriels
[Yan 04]	raster, liste de points, vecteur, arc	graphe relationnel attribué	squelette	reconnaissance de symboles

TAB. 1.2 – Représentations des connaissances graphiques (2)

D'autres systèmes ([Pasternak 93], [Kiyko 95], [Ah-Soon 01], [Coüasnon 01], [Weindorf 02]) ont recours à des formalismes opérationnels. Ceux-ci sont basés sur des langages à base de spécification de contraintes et de grammaire. La partie formelle de ces langages opérationnels correspond alors à une structure comparable à un graphe relationnel attribué. Cette structure est produite durant le processus d'opérationnalisation de ces langages. Plusieurs systèmes ([Joseph 92], [Wu 93], [Wenyin 99], [Song 02]) ont recours à un formalisme orienté objet. Les primitives graphiques correspondent alors aux objets du langage. L'opérationnalisation des objets au sein des systèmes simplifie la gestion des primitives graphiques. L'utilisation d'un tel formalisme permet également d'exploiter les relations d'héritage entre objets dans des buts d'extensibilité et de ré-utilisabilité.

À partir des formalismes les systèmes représentent leurs connaissances graphiques selon des modèles donnés. L'étude de ces différents modèles permet de constater différents points. D'abord ceux-ci sont souvent peu ou mal formalisés dans la littérature. Ensuite, chaque système utilise un modèle de représentation figé fonction des besoins de son application. Ce modèle est alors défini en fonction de l'expertise du concepteur du système : il est souvent peu justifié. La figure (1.14) illustre ces différents points à travers deux exemples de modèles. Ces modèles sont extraits des systèmes de [Han 94] et [Ramel 00] référencés dans nos tableaux comparatifs. Leurs constructions sont basées sur une étape préliminaire de vectorisation des contours (a). [Han 94] (b) construit à partir de ces contours vectorisés deux types de relation : connexion et adjacence. De cette façon son modèle décrit les relations existantes entre contours connectés et opposés. [Ramel 00] (c) construit son modèle à l'aide d'une étape supplémentaire d'appariement de contours. Différents quadrilatères sont alors construits en recherchant les meilleures relations de colinéarité et de parallélisme entre vecteurs. À partir de ces quadrilatères [Ramel 00] construit différentes relations structurelles : angle ( $L$ ), jonction ( $T$ ), parallélisme ( $//$ ), ... De cette façon, les vecteurs décrits par des relations d'adjacence dans le modèle de [Han 94] sont assimilés aux quadrilatères dans le modèle de [Ramel 00]. À l'opposé, les relations  $\{T, L\}$  décrites dans le modèle de [Ramel 00] sont assimilables aux relations de connexion dans le modèle de [Han 94].

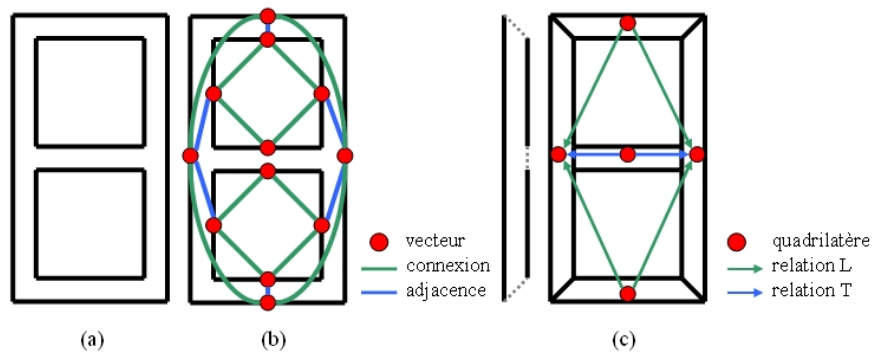


FIG. 1.14 – (a) contours vectorisés (b) modèle de [Han 94] (c) modèle de [Ramel 00]

Les modèles de [Han 94] [Ramel 00] sont donc proches. En effet ils décrivent tous les deux les formes à partir de leurs contours. Ils emploient également un formalisme commun (graphe relationnel attribué), des primitives graphiques similaires (vecteurs et quadrilatères) et des relations comparables (adjacence, connexion, relations  $\{,L,T,\dots\}$ ,  $\dots$ ). Les différences existantes entre ces deux modèles sont justifiées par les besoins des applications des systèmes qui les emploient. Celles-ci sont présentées sur la figure (1.15). [Han 94] applique son modèle pour à la squelettisation de documents graphiques (a). Il extrait pour cela des contours des vecteurs, mais aussi des cercles et des courbes. Celui-ci a alors défini son modèle afin que les connexités des graphes construits correspondent à celle des composantes connexes. De cette façon chaque graphe connexe de vecteurs dans son modèle correspond au squelette d'une composante dans l'image. [Ramel 00] applique son modèle à l'interprétation de plans cinématiques (b). Dans ces plans différentes relations peuvent être construites entre objets proches mais non connexes comme celles de parallélisme ( $//$ ). [Ramel 00] a donc défini son modèle afin d'intégrer ces relations permettant ainsi une meilleure représentation des objets graphiques présents dans les plans cinématiques.

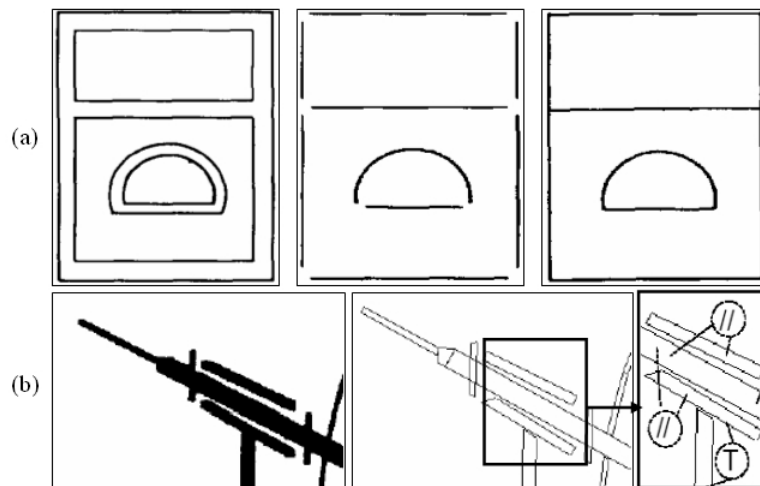


FIG. 1.15 – (a) application de [Han 94] (b) application de [Ramel 00]

Dans les tableaux (1.1) et (1.2) nous avons regroupé les modèles en différentes classes principales. En effet, en raison de la spécificité de chacun des modèles et leur manque d'explicitation leur définition exhaustive sortirait du cadre de ce manuscrit. De plus on peut dégager des points communs entre les différents modèles employés. Ces derniers définissent avant tout des relations entre des primitives graphiques données, ils sont donc dépendants de ces dernières. Nous avons regroupé ces modèles en trois classes : région, contour, et squelette. Celles-ci correspondent à différentes perceptions (et compositions) graduelles des objets graphiques de l'extérieur vers l'intérieur. Nous les illustrons sur la figure (1.16) à travers des exemples de modèles de différents systèmes. Nous les détaillons dans la suite de cette section.

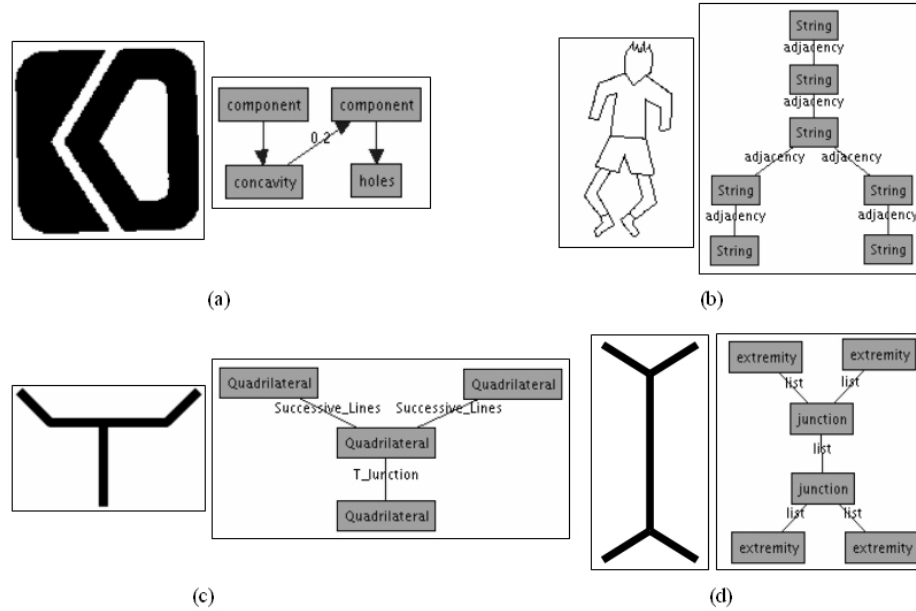


FIG. 1.16 – région (a) voisinage/inclusion [Badawy 02] (b) adjacence [Lladós 01]  
(c) contour [Ramel 00] (d) squelette [Popel 02]

La classe région concerne les modèles décrivant des relations entre primitives graphiques de type composante. On distingue deux types de modèle région dans la littérature. Le premier type exploite des relations d'inclusion et/ou de voisinage entre composantes. Ce type de modèle est utilisé dans différents systèmes : [Badawy 02], [Burge 98], [Leplumey 02], ... La figure (1.16) (a) donne un exemple des travaux de [Badawy 02]. Dans ce modèle une occlusion (*h*) est incluse dans une composante (*o*). Une relation de voisinage, et plus particulièrement de concavité, est décrite entre deux composantes (*o*) à travers un noeud et un arc attribués respectivement (c) et (0.2). L'autre type de modèle exploite les relations d'adjacence entre régions. Celles-ci sont alors construites à partir de l'analyse des frontières communes entre occlusions d'un squelette. Ce modèle est basé le plus souvent sur un formalisme de graphe relationnel attribué, on parle de graphe d'adjacence de région<sup>33</sup>. Ce type de modèle est utilisé dans de nombreux systèmes : [Seong 93], [Lladós 01], [Weindorf 02], [Barbu 04], ... La figure (1.16) (b) donne un exemple des travaux de [Lladós 01]. Dans ce modèle les régions closes d'un squelette vectorisé ont été reconstruites sous forme de chaînes (*String*). Les arcs décrivent alors des relations d'adjacence (*Adjacency*) entre les chaînes. À travers cet exemple on voit qu'une relation d'adjacence décrit une relation de voisinage entre occlusions d'une même composante connexe. Les modèles basés sur l'adjacence de régions sont donc très proches de ceux basés sur les relations d'inclusion et/ou de voisinage : l'adjacence peut être vue comme une relation de voisinage entre régions connexes [Clementini 93].

<sup>33</sup>Region Adjacency Graph (RAG)

Les modèles des classes contour et squelette ont tous les deux pour but de décrire les axes médians. Ils sont donc très proches, leur différence réside dans la nature des primitives graphiques utilisées qui sont respectivement issues des contours et du squelette. Ces modèles sont largement utilisés dans différents systèmes : [Joseph 92], [Han 94], [Kiyko 95], [Lee 98], [Ramel 00], [Popel 02], [Hilaire 04], ... La figure (1.16) (c) donne un exemple de modèle contour de [Ramel 00]. Dans ce modèle les contours ont été vectorisés et appariés sous forme de quadrilatère (*Quadrilateral*). Différentes relations sont décrites entre ces quadrilatères de type jonction (*T\_Junction*) ou simple connexion (*Successive\_Lines*). La figure (1.16) (d) donne un exemple de modèle squelette de [Popel 02]. Dans ce modèle les noeuds représentent les pixels jonctions (*Junction*) et extrémités (*Extremity*) du squelette, et les arcs les listes de points (*list*) les reliant.

Nous avons comparé ici diverses représentations des connaissances graphiques employées dans la littérature. Il existe tout d'abord de nombreuses similitudes entre les formalismes employés (primitives graphiques et formalismes à base de graphes). Les systèmes représentent à partir de ces formalismes leurs connaissances graphiques selon des modèles donnés. Ceux-ci se décomposent en différentes classes principales correspondant à différentes perceptions et compositions des objets graphiques, des régions vers les squelettes. Les systèmes manipulent donc des connaissances graphiques a priori très proches, cependant chaque système emploie un modèle qui lui est propre en fonction des besoins de son application. Nous en concluons que l'échange des connaissances graphiques entre les systèmes et leurs opérateurs est avant tout un problème d'interopérabilité sur les modèles. Celle-ci concerne l'échange de connaissances graphiques a priori proches (en termes de primitives graphiques et de classes de modèles) entre différents opérateurs malgré les divergences de modélisation. Elle peut donc être considérée comme une forme simplifiée d'interopérabilité sémantique. Dans la section suivante nous présentons la notion de multi-représentation, et illustrons comment cette notion permet de répondre à la problématique de l'interopérabilité sur les modèles.

### 1.3.5 Multi-représentation

La multi-représentation est une problématique de recherche mise à jour par [Wah 89] et [Minsky 91]. [Wah 89] dans son état de l'art conclut sur : "l'absence de techniques pour guider dans l'évaluation et la sélection d'une représentation des connaissances"<sup>34</sup>. [Minsky 91] déduit lui de la comparaison des approches symbolique et connexionniste que : "Pour résoudre des problèmes difficiles, nous devons utiliser plusieurs représentations différentes. Ceci car chaque type de structure de données a ses propres vertus et déficiences, et aucune ne semble adéquate en elle même pour toutes les fonctions relevant de ce que nous nommons le sens commun"<sup>35</sup>.

<sup>34</sup>"The lack of techniques to guide the evaluation and selection of a knowledge representation".

<sup>35</sup>"To solve really hard problems, we'll have to use several different representations. This is because each particular kind of data structure has its own virtues and deficiencies, and none by itself seems adequate for all the different functions involved with what we call common sense".

Aujourd'hui la multi-représentation est une problématique émergente dans divers domaines d'application comme la vision [Messina 01], le data mining [Kailing 04a], la simulation [Lara 04] ou l'indexation d'image [Kailing 04b]. En analyse des documents graphiques il n'existe selon notre état de l'art aucuns travaux sur cette question. Nous sommes les premiers à avoir introduit cette problématique au travers de nos publications<sup>36</sup>. Malgré cette émergence il n'existe pas de consensus sur cette notion de multi-représentation. Nous en proposons la définition (9) basée sur nos différents travaux.

**Définition 9 *Multi-représentation* :**

*La multi-représentation est la capacité d'un système à faire varier des représentations de connaissances (formalismes et/ou modèles), connues a priori, ainsi qu'à les évaluer et à les sélectionner afin de les adapter au mieux à ses besoins.*

Dans notre état de l'art nous nous limiterons aux aspects "représentation" de la multi-représentation : c'est à dire les variations des formalismes et/ou modèles. En effet les aspects "évaluation/sélection" sont du domaine de l'opérationnalisation des connaissances. Ceux-ci sont d'ailleurs centraux dans la discipline de l'extraction de connaissances<sup>37</sup> [Fayyad 96]. Les différentes représentations des connaissances sont directement en relation avec les algorithmes qui les manipulent [Messina 01] : elles sont donc connues a priori par ces algorithmes. Dans le cadre des connaissances graphiques ces représentations sont construites par les algorithmes aux travers des opérateurs d'extraction. Selon notre point de vue on peut distinguer deux types principaux de relation de construction : 1 algorithme vers  $n$  représentations et  $n$  algorithmes vers  $n$  représentations. Nous détaillons ces deux relations par la suite.

Dans la première relation (1 –  $n$ ) un même algorithme (ou des algorithmes proches) est utilisé pour la construction de différentes représentations. La figure (1.17) donne deux exemples de représentations différentes de type graphe (non-orienté et orienté) obtenues à partir d'algorithmes de squelettisation [Popel 02] [Lin 02]. Dans le graphe de squelette non-orienté (a) [Popel 02], les noeuds représentent les pixels jonctions et extrémités du squelette, et les arcs les listes de points entre les noeuds. Les noeuds ( $j_i$ ) sont étiquetés selon leur degré de connexité ( $i$ ) (1 pour les pixels extrémités,  $\geq 3$  pour les pixels jonctions). Dans le graphe de squelette orienté (b) [Lin 02], les noeuds représentent les chaînes de pixels, et les arcs orientés des relations angulaires symboliques entre ces chaînes. Plus précisément, pour chaque chaîne un arc est construit avec sa chaîne la plus proche et telle que l'angle les séparant soit le plus proche de ( $\pi$ ). À travers ces deux graphes, nous avons deux points de vue complémentaires de représentation d'un squelette. Le premier (a) est centré sur la représentation des jonctions du squelette (a), tandis que le second (b) est centré sur la représentation des chaînes du squelette. Chacune des représentations est utilisée à des buts différents : compression de données (a) et reconnaissance de caractère script (b).

<sup>36</sup>Nous reportons le lecteur page ?? pour le détail de ces publications.

<sup>37</sup>Knowledge Discovery

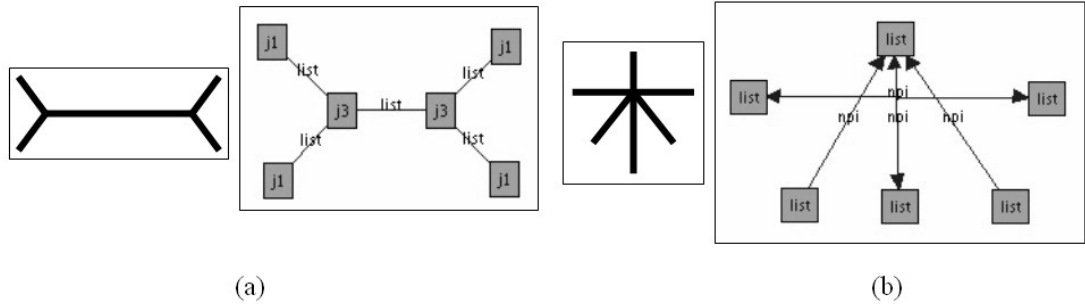


FIG. 1.17 – (a) représentation jonction (b) représentation chaîne

Dans la seconde relation ( $n-n$ ) différents algorithmes sont utilisés pour la construction de différentes représentations. Il s'agit de la relation permettant le plus de variabilité en matière de multi-représentation. En effet, elle concerne à la fois la variation des primitives graphiques ainsi que des modèles structurant ces primitives. La figure (1.18) donne deux exemples de représentations obtenues à partir d'algorithmes à base de décomposition en plages (a) [Burge 98] et de squelettisation (b) [Yan 03]. Dans l'algorithme à base de décomposition en plages (a) [Burge 98] exploite des primitives graphiques de type région. Ces primitives représentent respectivement des régions jonctions ( $N_a$ )<sup>38</sup>, lignes ( $E_a$ )<sup>38</sup>, et extrémités ( $E$ ). Elles sont obtenues à la suite de la concaténation de plages verticales et horizontales. Elles servent par la suite d'attributs répartis à travers les noeuds  $\{N_a, E\}$  et les arcs  $\{E_a\}$  d'un graphe. Dans l'algorithme à base de squelettisation (b) [Yan 03] représente le squelette par un ensemble de vecteurs ( $V_i$ ). Ceux-ci sont obtenus à la suite d'une étape de chaînage du squelette et de polygonalisation. Les vecteurs constituent les noeuds d'un graphe reliés par un ensemble d'arcs  $\{C, P\}$  décrivant des relations<sup>39</sup> de connexion ( $C$ ) et de parallélisme ( $P$ ) entre les vecteurs.

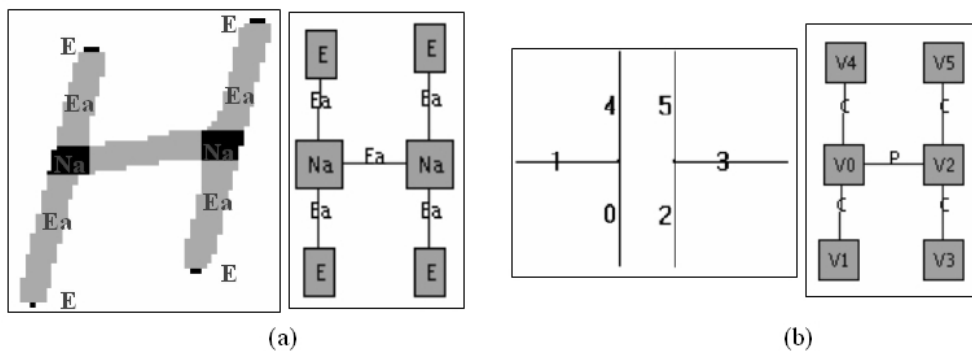


FIG. 1.18 – (a) représentation en régions (b) représentation en vecteurs

<sup>38</sup>Node and Edge area

<sup>39</sup>[Yan 03] ne décrit qu'une partie des relations pour des raisons de complexité.

Ces relations  $(1 - n)$  et  $(n - n)$  illustrent les possibilités de multi-représentation des formes graphiques. Un système peut donc exploiter différentes représentations afin de les adapter au mieux aux formes graphiques traitées. Cependant la mise en oeuvre de la multi-représentation au sein d'un système soulève la problématique de gestion de différentes représentations par les opérateurs. Nous illustrons cette dernière sur la figure (1.19) (a). Celle-ci consiste pour les opérateurs ( $Op_n$ ) d'un système à manipuler différentes représentations ( $R_n$ ) connues a priori : elle impose donc l'utilisation d'un formalisme commun (ou pivot) entre ces opérateurs. Cette problématique est en fait complémentaire à celle de l'interopérabilité sémantique introduite précédemment. Cette dernière est illustrée sur la figure (1.19) (b). Elle concerne l'échange de connaissances graphiques entre opérateurs ( $Op_n$ ) (de un ou de différents systèmes) exploitant différentes représentations ( $R_n$ ) inconnues a priori. On peut donc en conclure qu'il ne peut donc y avoir d'interopérabilité sémantique sans prise en compte de la multi-représentation. Ces deux problématiques sont complémentaires et se combinent naturellement comme nous l'illustrons sur la figure (1.19) (c). Des systèmes inter-opérables peuvent donc être vus comme des systèmes à base de multi-représentation. Chacun des opérateurs vient alors rechercher (ou "reconnaître") des représentations ( $R_n$ ) qu'il est susceptible de pouvoir manipuler afin d'en extraire des connaissances graphiques.

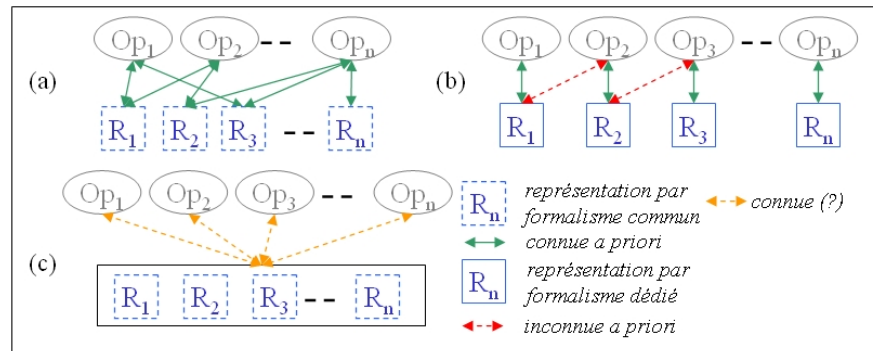


FIG. 1.19 – (a) multi-représentation (b) interopérabilité sémantique  
(c) multi-représentation & interopérabilité sémantique

### 1.3.6 Conclusion

Dans cette section nous avons présenté un état de l'art sur la gestion des connaissances graphiques. Notre approche dans ce manuscrit vise à la combinaison des méthodes d'extraction de primitives graphiques. Cependant cette combinaison soulève la problématique de l'interopérabilité entre opérateurs d'extraction. Cette dernière concerne l'échange de connaissances graphiques représentées de différentes façons entre les opérateurs. Afin de répondre à cette problématique nous avons plus particulièrement orienté notre état de l'art sur l'aspect représentation des connaissances graphiques (formalismes et modèles).



Nous avons présenté une décomposition des connaissances graphiques en deux niveaux de formalismes : primitives graphiques et formalismes à base de graphes. Les primitives graphiques se décomposent elles en formalismes à base de pixels et à base de vecteurs. Dans ces deux cas de formalismes les différentes primitives graphiques répondent à de fortes relations de composition et de spécialisation. Les formalismes à base de graphes (réseaux sémantiques, graphes conceptuels, schémas, langages O.O et graphes relationnels attribués) permettent eux de décrire les relations entre les primitives graphiques afin de représenter les objets graphiques de plus haut niveau.

Nous avons ensuite comparé différentes représentations des connaissances graphiques (formalismes et modèles) employées au sein des systèmes dans la littérature. Il existe tout d'abord de nombreuses similitudes entre les formalismes employés (primitives graphiques et formalismes de plus haut-niveau à base de graphes). À partir de ces formalismes les systèmes représentent leurs connaissances graphiques selon des modèles donnés. Ceux-ci se décomposent en différentes classes principales correspondant à différentes perceptions et compositions des objets graphiques, des régions vers les squelettes. Cependant chaque système emploie un modèle qui lui est propre fonction des besoins de son application. Les systèmes manipulent donc des connaissances graphiques a priori très proches, mais chacun des modèles reste spécifique à un système donné. Nous en avons conclu que l'échange des connaissances graphiques entre opérateurs correspondait à la problématique de l'interopérabilité sur les modèles. Cette dernière concerne l'échange de connaissances graphiques a priori proches (en termes de primitives graphiques et de classes de modèles) entre différents opérateurs malgré les divergences de modélisation. Elle peut donc être considérée comme une forme simplifiée d'interopérabilité sémantique. Nous avons ensuite présenté la notion de multi-représentation et comment celle-ci répondait à la problématique de l'interopérabilité sémantique. Celle-ci traduit principalement la capacité d'un système à faire varier des représentations de connaissances (formalismes et/ou modèles) connues a priori. Cette capacité de variation repose alors sur l'utilisation d'un formalisme commun (ou pivot). Nous avons alors montré que dans des systèmes inter-opérables une approche à base de multi-représentation devait être exploitée. Chacun des opérateurs de ces systèmes vient alors rechercher ou reconnaître les différentes représentations qu'il est susceptible de manipuler afin d'en extraire des connaissances graphiques.

## 1.4 Conclusion

Dans ce chapitre nous avons abordé le problème de l'échange des primitives graphiques entre les opérateurs d'extraction pour leur combinaison. Nous avons formulé ce problème comme un problème de gestion des connaissances dites "graphiques" au sein des systèmes d'analyse des documents.

Nous avons présenté en première partie un état de l'art sur la gestion des connaissances à travers ses concepts généraux : représentation, formalisme, modèle, langage de représentation et opérationnalisation. À l'issue de cet état de l'art nous avons soulevé le problème de l'interopérabilité, en particulier sémantique, entre systèmes. Celle-ci concerne les problèmes de manipulation et d'échange des connaissances entre différents systèmes malgré les divergences de représentations.

Dans la seconde partie nous avons présenté un état de l'art sur la gestion des connaissances graphiques. De façon à répondre au problème de l'interopérabilité entre opérateurs d'extraction nous avons abordé dans cet état de l'art l'aspect représentation des connaissances graphiques (formalismes et modèles). Cet état de l'art illustre premièrement les fortes relations de composition et de spécialisation des connaissances graphiques. Elles sont représentées graduellement au sein des systèmes du niveau de représentation image jusqu'à celui de symbole. Il illustre ensuite les fortes similitudes existantes entre les différents systèmes en termes de formalismes et classes de modèles employés. Il conclut alors que l'échange des connaissances graphiques entre opérateurs est un problème d'interopérabilité sur les modèles. Cette dernière peut être considérée comme une forme simplifiée d'interopérabilité sémantique. Nous avons alors introduit la notion de multi-représentation et comment celle-ci répondait à la problématique de l'interopérabilité sémantique. Dans les systèmes inter-opérables chacun des opérateurs vient alors rechercher ou reconnaître les différentes représentations qu'il est susceptible de manipuler afin d'en extraire des connaissances graphiques.

À l'issue des états de l'art présentés dans ce chapitre nous concluons sur deux points principaux. Premièrement que les connaissances graphiques répondent à de fortes relations de composition et de spécialisation. Deuxièmement que l'échange des connaissances graphiques relèvent de deux problématiques principales : l'interopérabilité sur les modèles, et la multi-représentation. À la vue de ces deux conclusions nous proposons de formaliser la combinaison des opérateurs d'extraction comme un processus reconstruction d'objets. Nous développons cette approche dans notre partie contributions qui suit.

# Bibliographie

- [Ah-Soon 01] C. Ah-Soon & K. Tombre. *Architectural Symbol Recognition Using a Network of Constraints*. Pattern Recognition Letters (PRL), vol. 22, no. 2, pages 231–248, 2001.
- [Ahmadyfard 00] A. Ahmadyfard & J. Kittler. *Region-Based Object Recognition : Pruning Multiple Representations and Hypotheses*. In British Machine Vision Conference (BMVC), pages 745–754, 2000.
- [Ahmed 00] M. Ahmed & R. Kreidieh. *An Expert System for General Symbol Recognition*. Pattern Recognition (PR), vol. 33, no. 12, pages 1975–1988, 2000.
- [André 01] P. André & A. Vailly. *Conception des systèmes d'information*. Editions Ellipses, ISBN : 272980479X, 2001.
- [Bachimont 94] B. Bachimont. *Le controle dans les systèmes à base de connaissances*. Editions Hermès, ISBN : 2-86601-425-1, 1994.
- [Bachimont 04] B. Bachimont. *Arts et Sciences du Numérique : Ingénierie des Connaissances et Critique de la Raison Computationnelle*. Habilitation à Diriger les Recherches, Université de Technologie de Compiègne, France, 2004.
- [Badawy 02] O. El Badawy & M. Kamel. *Shape Representation Using Concavity Graphs*. In International Conference on Pattern Recognition (ICPR), pages 461–464, 2002.
- [Bahr 02] H.P. Bahr & H. Lenk. *Steps to Cognition in Image Analysis and in Philosophy*. In Symposium on Photogrammetric Computer Vision (PCV), volume B, pages 16–18, 2002.
- [Barbu 04] E. Barbu, P. Héroux, S. Adam & E. Trupin. *Frequent Graph Discovery : Application to Line Drawing Document Images*. Electronic Letters on Computer Vision and Image Analysis (ELCVIA), vol. 5, no. 2, pages 45–47, 2004.
- [Barr 89] A. Barr, P.R. Cohen & E.A. Feigenbaum. *The handbook of artificial intelligence*, volume 1-4. Addison Wesley Publisher, ISBN : 0201118106, 1989.
- [Barrow 71] H.G. Barrow & R.J. Popplestone. *Relational Descriptions in Picture Processing*. Machine Intelligence, vol. 6, pages 377–396, 1971.
- [Barry 01] C. Barry, C. Cormier, G. Kassel & J. Nobécourt. *Évaluation de Langages Opérationnels de Représentation d'Ontologies*. In Conférence sur l'Ingénierie des Connaissances (IC), pages 309–327, 2001.
- [Barthélemy 05] M. Barthélemy, E. Chow & T. Eliassi-Rady. *Knowledge Representation Issues in Semantic Graphs for Relationship Detection*. In AAAI Spring Symposium, pages 91–98, 2005.
- [Berge 83] C. Berge. *Graphes*. Gauthier-Villards, 3 edition, ISBN : 2-04-015555-4, 1983.

- [Bobrow 77] D. Bobrow & T. Winograd. *An Overview of KRL, a Knowledge Representation Language*. Cognitive Science, vol. 1, no. 1, pages 3–45, 1977.
- [Burge 95] M. Burge & G. Monagan. *Using the Voronoi Tessellation for Grouping Words and Multipart Symbols in Document*. In Conference on Vision Geometry IV, pages 116–124, 1995.
- [Burge 98] M. Burge & W.G. Kropatsh. *A Minimal Line Property Preserving Representation of Line Images*. In Conference on Structural and Syntactical Pattern Recognition (SSPR), volume 1451 of *Lecture Notes in Computer Science (LNCS)*, pages 355–368, 1998.
- [Charlet 02] J. Charlet. *L'Ingénierie des Connaissances : Développements, Résultats, et Perspectives pour la Gestion des Connaissances Médicales*. Habilitation à Diriger les Recherches, Université Pierre et Marie Curie, France, 2002.
- [Chen 04] Y. Chen, J. Gong, W. Jia & Q. Zhang. *XML-based Spatial Data Interoperability on The Internet*. In Conference of International Society for Photogrammetry and Remote Sensing and Spatial Information Sciences (ISPRS), pages 167–201, 2004.
- [Chhabra 98] A. Chhabra. *Graphic Symbol Recognition : An Overview*. In Workshop on Graphics Recognition (GREC), volume 1389 of *Lecture Notes in Computer Science (LNCS)*, pages 68–79, 1998.
- [Clementini 93] E. Clementini, P.D. Felice & P.V. Oosterom. *A Small Set of Formal Topological Relationships Suitable for End-User Interaction*. In Symposium on Advances in Spatial Databases (SSD), volume 692 of *Lecture Notes In Computer Science (LNCS)*, pages 277–295, 1993.
- [Coüasnon 01] B. Coüasnon. *DMOS : a Generic Document Recognition Method, Application to an Automatic Generator of Musical Scores, Mathematical Formulae and Table Structures Recognition Systems*. In International Conference on Document Analysis and Recognition (ICDAR), pages 215–220, 2001.
- [Collin 93] S. Collin, K. Tombre & P. Vaxivière. *Don't Tell Mom I'm Doing Document Analysis ; She Believes I'm in the Computer Vision Field*. In International Conference on Document Analysis and Recognition (ICDAR), pages 619–622, 1993.
- [Cordella 00] L.P. Cordella & M. Vento. *Symbol Recognition in Documents : a Collection of Techniques*. International Journal on Document Analysis and Recognition (IJ DAR), vol. 3, no. 2, pages 73–88, 2000.
- [Defude 05] B. Defude. *Bases de Données : de l'Objet à l'Interopérabilité*. Habilitation à Diriger des Recherches, Université Pierre et Marie Curie, Paris, France, 2005.
- [Doermann 98] D. Doermann. *The Indexing and Retrieval of Document Images : a Survey*. Rapport technique CS-TR-3876, University of Maryland, USA, 1998.
- [Dunlop 95] M.D. Dunlop & I.G. Ruthven. *An Analysis of the Use of Graphics for Information Retrieval*. In WorkShop on Multimedia Information Retrieval (MIRO), 1995.
- [Ermine 96] J.L. Ermine. *Les systèmes de connaissances*. Editions Hermès, 1996.
- [Euzenat 98] J. Euzenat. *Représentation des Connaissances par Objets*. In Langages et Modèles à Objets : Etat et Perspectives de la Recherche, pages 293–319. Editions INRIA, ISBN : 2-7261-1131-9, 1998.

- [Euzenat 99] J. Euzenat. *Représentations de connaissance : de l'Approximation à la Confrontation*. Habilitation à Diriger des Recherches, Université Joseph Fourier, Grenoble, France, 1999.
- [Fan 98] K.C. Fan, D.F. Chen & M.G. Wen. *Skeletonization of Binary Images with Nonuniform Width via Block Decomposition and Contour Vector Matching*. Pattern Recognition (PR), vol. 31, no. 7, pages 823–838, 1998.
- [Farreny 89] H. Farreny. *Les systèmes experts, principes et exemples*. Editions Cepadues, ISBN : 2854281306, 1989.
- [Fayyad 96] U. Fayyad, G. Piatetsky Shapiro & P. Smyth. *From Data Mining to Knowledge Discovery in Databases*. Artificial Intelligence (AI), vol. 17, pages 37–54, 1996.
- [Foggia 99] P. Foggia, C. Sansone, F. Tortorella & M. Vento. *Definition and Validation of a Distance Measure Between Structural Primitives*. Pattern Analysis and Applications (PAA), vol. 2, no. 3, pages 215–227, 1999.
- [Fonseca 04] M.J. Fonseca, B. Barroso, P. Ribeiro & J.A. Jorge. *Retrieving Vector Graphics Using Sketches*. In Symposium on Smart Graphics (SG), volume 3031 of *Lecture Notes in Computer Science (LNCS)*, pages 66–76, 2004.
- [Fürst 04] F. Fürst. *Contribution à l'Ingénierie des Ontologies : une Méthode et un Outil d'Operationalisation*. Thèse de Doctorat, Université de Nantes, France, 2004.
- [Gandon 02] F. Gandon. *Ontology Engineering : a Survey and a Return on Experience*. Rapport technique 4396, INRIA Sophia Antipolis, France, 2002.
- [Gensel 92] J. Gensel. *Contraintes et Représentation de Connaissances par Objets Application au Modèle Tropes*. Thèse de Doctorat, Université de Joseph Fourier, Grenoble, France, 1992.
- [Gordon 97] M.B. Gordon. *Sciences cognitives : Diversité des approches*. Editions Hermès, ISBN : 1-892512-06-8, 1997.
- [Gould 99] M. Gould. *Experiments Using VML : An Emerging Vector Graphics Standard*. In Agile Conference on Geographic Information Science, 1999.
- [Gross 03] J.L. Gross & J. Yellen. *Handbook of graph theory*. CRC Press, ISBN : 1584880902, 2003.
- [Gruber 95] T.R. Gruber. *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. International Journal of Human Computer Studies (IJHCS), vol. 43, no. 5/6, pages 907–928, 1995.
- [Han 94] C.C. Han & K.C. Fahn. *Skeleton Generation of Engineering Drawings via Contour Matching*. Pattern Recognition (PR), vol. 27, no. 2, pages 261–275, 1994.
- [Hartog 96] J.E. Den Hartog. *Knowledge Based Interpretation of Utility Maps*. Computer Vision and Image Understanding (CVIU), vol. 63, no. 1, pages 105–117, 1996.
- [Haton 91] J.P. Haton & al. *Le raisonnement en intelligence artificielle*. InterEditions, ISBN : 2-7296-0335-2, 1991.
- [Hentenryck 89] P.V. Hentenryck. *Constraint satisfaction in logic programming*. MIT Press, ISBN : 0-262-08181-4, 1989.
- [Herman 02] I. Herman & D. Dardailler. *SVG Linearization and Accessibility*. Computer Graphics Forum, vol. 21, no. 4, pages 777–786, 2002.

- [Hilaire 04] X. Hilaire. *Segmentation Robuste de Courbes Discrètes 2D et Applications à la Retroconversion de Documents Techniques*. Thèse de Doctorat, Institut National Polytechnique de Lorraine (INRIA), France, 2004.
- [Holsapple 04] C.W. Holsapple. Handbook on knowledge management, volume 1-2. Springer Verlag Publisher, 2004.
- [Huang 97] R.W. Huang. *Indexing Pictures by Key Objects for Large-Scale Image Databases*. Pattern Recognition (PR), vol. 30, no. 7, pages 1229–1237, 1997.
- [Joseph 92] S.H. Joseph & T.P. Pridmore. *Knowledge-Directed Interpretation of Line Drawing Images*. Pattern Analysis and Machine Intelligence (PAMI), vol. 14, no. 9, pages 928–940, 1992.
- [Kailing 04a] K. Kailing, H.P. Kriegel, A. Pryakhin & M. Schubert. *Clustering Multi-Represented Objects with Noise*. In Conference on Knowledge Discovery and Data Mining (PKAD), volume 3056 of *Lecture Notes in Computer Science (LNCS)*, pages 394–403, 2004.
- [Kailing 04b] K. Kailing, H.P. Kriegel & S. Schonauer. *Content-Based Image Retrieval Using Multiple Representations*. In Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES), volume 3214 of *Lecture Notes in Computer Science (LNCS)*, pages 982–988, 2004.
- [Kasturi 88] R. Kasturi & J. Alemany. *Information Extraction from Images of Paper-Based Maps*. Software Engineering, vol. 14, no. 5, pages 671 – 675, 1988.
- [Kayser 97] D. Kayser. La représentation des connaissances. Editions Hermès, ISBN : 2-86601-647-5, 1997.
- [Kernighan 99] B.W. Kernighan & P. Pike. The practice of programming. Addison Wesley Publishing, ISBN : 0-201-61586-X, 1999.
- [Kiyko 95] V.M. Kiyko. *Recognition of Objects in Images of Paper Based Line Drawings*. In International Conference on Document Analysis and Recognition (ICDAR), pages 970–975, 1995.
- [Klages 50] L. Klages. Les principes de la caractérologie. Editions Delachaux et Niestlé, 1950.
- [Kolodner 93] J. Kolodner. Case-based reasoning. Morgan Kaufmann Editor, ISBN : 1-55860-237-2, 1993.
- [Kuipers 94] B. Kuipers. *Algernon for Expert Systems*. Rapport technique, Computer Science Department, University of Texas at Austin, USA, 1994.
- [Lara 04] J. Lara, H. Vangheluwe & M. Alfonseca. *Meta-Modelling and Graph Grammars for Multi-Paradigm Modelling in AToM*. Software and Systems Modelling (SoSyM), vol. 3, no. 3, pages 194–209, 2004.
- [Lassila 90] O. Lassila. *Frames or Objects, or Both?* Rapport technique HTKK-TKO-B67, Department of Computer Science, Helsinki University of Technology, Finland, 1990.
- [Lee 98] K.H. Lee, S.B. Cho & Y.C. Choy. *A Knowledge-Based Automated Vectorizing System for Geographic Information System*. In International Conference on Pattern Recognition (ICPR), volume 2, pages 1546–1548, 1998.
- [Leondes 00] C. Leondes. Knowledge-based systems, volume 1-4. Academic Press, 2000.
- [Leplumey 02] I. Leplumey & Ch. Queguiner. *Un Graphe de Voisinage Basé sur l'Utilisation des Distances Discrètes*. In Colloque International Francophone sur l'Écrit et le Document (CIFED), pages 41–50, 2002.

- [Lin 02] F. Lin & X. Tang. *Off-Line Handwritten Chinese Character Stroke Extraction*. In International Conference on Pattern Recognition (ICPR), volume 3, pages 249–252, 2002.
- [Lladós 01] J. Lladós, E. Marti & J.J. Villuanueva. *Symbol Recognition by Error Subgraph Matching Between Region Adjacency Graphs*. Pattern Analysis and Machine Intelligence (PAMI), vol. 23, no. 10, pages 1137–1143, 2001.
- [Lladós 02] J. Lladós, E. Valveny, G. Sánchez & E. Martí. *Symbol Recognition : Current Advances and Perspectives*. In Workshop on Graphics Recognition (GREC), volume 2390 of *Lecture Notes in Computer Science (LNCS)*, pages 104–127, 2002.
- [Lorenz 95] O. Lorenz. *Automatic Indexing for Storage and Retrieval of Line Drawings*. In Storage and Retrieval for Media Databases, volume 2420 of *SPIE Proceedings*, pages 216–227, 1995.
- [Love 01] D.M. Love & J.A. Barton. *Drawing Retrieval Using an Automated Coding Technique*. In Conference on Flexible Automation and Intelligent Manufacturing (FAIM), pages 158–166, 2001.
- [Lucas 86] M. Lucas. *Algorithmique et representation des donnees*, volume 2. Editions Masson, ISBN : 2-225-80924-0, 1986.
- [Mahmood 95] T.F.S. Mahmood. *Indexing of Technical Manual Document Databases*. In Storage and Retrieval for Image and Video Databases, volume 2420 of *SPIE Proceedings*, pages 430–441, 1995.
- [Mark 89] D. Mark & al. *Working Bibliography on "Languages of Spatial Relations"*. Rapport technique 89-10, National Center for Geographic Information and Analysis, Univeristy of California, Santa Barbara, USA, 1989.
- [Mejbri 02] E.F. El Mejbri, H. Grabowski, H. Kunze, R.S. Lossack & A. Michelis. *3D Reconstruction of Paper Based Assembly Drawings : State of the Art and Approach*. In Workshop on Graphics Recognition (GREC), volume 2390 of *Lecture Notes in Computer Science (LNCS)*, pages 1–12, 2002.
- [Messina 01] E. Messina, J. Evans & J. Albus. *Evaluating Knowledge and Representation for Intelligent Control*. In Workshop on Performance Metrics (PerMIS), volume 982 of *NIST Special Publication*, 2001.
- [Messmer 95] B.T. Messmer. *Efficient Graph Matching Algorithms for Preprocessed Model Graphs*. Thèse de Doctorat, Bern University, Switzerland, 1995.
- [Milgram 93] M. Milgram. *Reconnaissance des formes, méthodes numériques et connexionnistes*. Editions Armand Collin, ISBN : 2200212909, 1993.
- [Minsky 75] M. Minsky. *A Framework for Representing Knowledge*. In P.H. Winston, editeur, *The Psychology of Computer Vision*, pages 211–277. McGraw-Hill, ISBN : 0070710481, 1975.
- [Minsky 91] M. Minsky. *Logical Versus Analogical or Symbolic Versus Connectionist or Neat Versus Scruffy*. AI Magazine, vol. 12, no. 2, pages 34–51, 1991.
- [Mori 92] S. Mori, C.Y. Suen & K. Yamamoto. *Historical Review of OCR Research and Development*. IEEE, vol. 80, no. 7, pages 1029–1058, 1992.
- [Nardelli 93] E. Nardelli, M. Fossa & G. Proietti. *Raster to Object Conversion Aided by Knowledge Based Image Processing*. In International Conference on Document Analysis and Recognition (ICDAR), pages 951–964, 1993.

- [Nygaard 63] K. Nygaard. *SIMULA - an Extension of ALGOL to the Description of Discrete-Event Networks*. In IFIP Congress, volume 62 of *Information Processing*, pages 520–522, 1963.
- [Okazaki 88] S. Okazaki & Y. Tsuji. *Knowledge-Based Approach for Adaptive Recognition of Drawings*. In International Conference on Pattern Recognition (ICPR), volume 301 of *Lecture Notes In Computer Science (LNCS)*, pages 627–632, 1988.
- [Otman 97] G. Otman. *Terminologie de l'intelligence artificielle*. La Maison du Dictionnaire de Paris, ISBN : 2-85608-087-1, 1997.
- [Pachet 97] François Pachet. *Représentation de Connaissances et Langages à Objets*. Habilitation à Diriger des Recherches, LIP6, Paris, France, 1997.
- [Pasternak 93] B. Pasternak & B. Neumann. *Adaptable Drawing Interpretation Using Object-Oriented and Constraint-Based Graphic Specification*. In International Conference on Document Analysis and Recognition (ICDAR), pages 359–364, 1993.
- [Pasternak 95] B. Pasternak & B. Neumann. *The Role of Taxonomy in Drawing Interpretation*. In International Conference on Document Analysis and Recognition (ICDAR), volume 2, pages 799–802, 1995.
- [Popel 02] D.V. Popel. *Compact Graph Model of Handwritten Images : Integration into Authentication and Recognition*. In Conference on Structural and Syntactical Pattern Recognition (SSPR), volume 2396 of *Lecture Notes in Computer Science (LNCS)*, pages 272–280, 2002.
- [Quillian 68] M.R. Quillian. *Semantic Memory*. In M. Minsky, editeur, *Semantic Information Processing*, pages 227–270. MIT Press, ISBN : 0262130440, 1968.
- [Ramel 00] J.Y. Ramel, N. Vincent & H. Emptoz. *A Structural Representation for Understanding Line-Drawing Images*. *International Journal on Document Analysis and Recognition (IJDA)*, vol. 3, pages 58–66, 2000.
- [Roberts 77] R.B. Roberts & I.P. Goldstein. *The FRL Manual*. Rapport technique AIM-409, Massachusetts Institute of Technology, Cambridge, USA, 1977.
- [Saidali 02] Y. Saidali. *Modélisation et Acquisition Incrémentale de Connaissances Traiteurs d'Images*. Thèse de Doctorat, Université de Rouen, France, 2002.
- [Sanfeliu 83] A. Sanfeliu & K.S. Fu. *A Distance Measure Between Attributed Relational Graph*. *Transactions on Systems, Man and Cybernetics (TSMC)*, vol. 13, no. 3, pages 353–362, 1983.
- [Sartipi 01] K. Sartipi & K. Kontogiannis. *A Graph Pattern Matching Approach to Software Architecture Recovery*. In International Conference on Software Maintenance (ICSM), pages 408–419, 2001.
- [Sciascio 04] E. Di Sciascio, F.M. Donini & M. Mongiello. *A Logic for SVG Documents Query and Retrieval*. *Multimedia Tools and Applications*, vol. 24, pages 125–153, 2004.
- [Seghrouchni 04] A.F. Seghrouchni, S. Haddad, T. Melitti & A. Suna. *Interopérabilité des Systèmes Multi-Agents à l'Aide des Services Web*. In Journées Francophones sur les Systèmes Mutli-Agents (JFSMA), pages 91–104, 2004.
- [Seguela 01] P. Seguela. *Construction de Modèle de Connaissances par Analyse Linguistique de Relations Lexicales dans les Documents Techniques*. Thèse de Doctorat, Université de Toulouse III, France, 2001.



- [Seong 93] D.S. Seong, H.S. Kim & K.H. Park. *Incremental Clustering of Attributed Graphs*. Transactions on Systems, Man and Cybernetics (TSMC), vol. 23, no. 5, pages 1399–1411, 1993.
- [Shih 89] C.C. Shih & R. Kasturi. *Extraction of Graphic Primitives from Images of Paper Based Line Drawings*. Machine Vision and Applications, vol. 2, pages 103–113, 1989.
- [Silvent 05] A.S. Silvent, C. Garbay, P.Y. Carry & M. Dojat. *Le Rôle des Données, Informations et Connaissances dans la Construction de Scénarios Médicaux*. Revue d’intelligence artificielle, vol. 19, pages 207–213, 2005.
- [Smeulders 00] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta & R. Jain. *Content-based Image Retrieval at the End of the Early Years*. Pattern Analysis and Machine Intelligence (PAMI), vol. 22, no. 12, pages 1349–1380, 2000.
- [Smith 01] G.J. Smith & M.L. Maher. *Knowledge Discovery in Architectural CAD Data*. <http://www.arch.usyd.edu.au/mary/kdd-research.html>, 2001.
- [Song 02] J. Song, F. Su, C. Tai & S. Cai. *An Object-Oriented Progressive-Simplification based Vectorization System for Engineering Drawings : Model, Algorithm and Performance*. Pattern Analysis and Machine Intelligence (PAMI), vol. 24, no. 8, pages 1048–1060, 2002.
- [Sowa 76] J.F. Sowa. *Conceptual Graphs for a Database Interface*. IBM Journal of Research and Development, vol. 20, no. 4, pages 336–357, 1976.
- [Sowa 99] J.F. Sowa. *Knowledge representation : Logical, philosophical, and computational foundations*. Cole Publishing Co, ISBN : 0-534-94965-7, 1999.
- [Spaccapietra 05] S. Spaccapietra. *Traitements de Connaissances*. Département d’Informatique, Laboratoire de Bases de Données, Ecole Polytechnique Fédérale de Lausanne, France, 2005.
- [Tombre 91] K. Tombre & P. Vaxiviere. *Structure, Syntax and Semantics in Technical Document Recognition*. In International Conference on Document Analysis (ICDAR), volume 1, pages 6149–1991, 1991.
- [Tombre 00] K. Tombre, C. Ah Soon, P. Dosch, G. Masini & S. Tabbone. *Stable and Robust Vectorization : How to Make the Right Choices*. In Workshop on Graphics Recognition (GREC), volume 1941 of *Lecture Notes in Computer Science (LNCS)*, pages 3–18, 2000.
- [Tombre 03] K. Tombre & B. Lamiroy. *Graphics Recognition - from Re-engineering to Retrieval*. In International Conference on Document Analysis and Recognition (ICDAR), pages 148–155, 2003.
- [Tsai 79] W.H. Tsai & K.S. Fu. *Error-Correcting Isomorphisms of Attributed Relational Graphs for Pattern Analysis*. Transactions on Systems, Man and Cybernetics (TSMC), vol. 9, no. 12, pages 757–768, 1979.
- [Tuthill 90] G.S. Tuthill. *Knowledge engineering : Concepts and practices for knowledge-based systems*. Tab Books, ISBN : 08-306-9297-5, 1990.
- [Ullman 89] J.D. Ullman. *Principles of database and knowledge base systems, volume 1*. Computer Sciences Press, ISBN : 0716781581, 1989.
- [Valencia 00] E. Valencia. *Gestion de l’Hétérogénéité Sémantique entre Agents Dialogiques à l’Aide d’Outils de Topologie Algébrique*. Thèse de Doctorat, Université Paris-XI, France, 2000.

- [Valtchev 99] P. Valtchev. *Construction Automatique de Taxonomies pour l'Aide à la Représentation de Connaissances par Objets*. Thèse de Doctorat, Université de Joseph Fourier, Grenoble, France, 1999.
- [Vaxiviere 94] P. Vaxiviere & K. Tombre. *Knowledge Organization and Interpretation Process in Engineering Drawing Interpretation*. In Workshop on Document Analysis Systems (DAS), pages 307–317, 1994.
- [Wah 89] B. Wah, M. Lowrie & G. Li. *Computers for Symbolic Processing*. IEEE, vol. 77, no. 4, pages 509–540, 1989.
- [Weindorf 02] M. Weindorf. *Structure Based Interpretation of Unstructured Vector Maps*. In Workshop on Graphics Recognition (GREC), volume 2390 of *Lecture Notes in Computer Science (LNCS)*, pages 190–199, 2002.
- [Wenyin 99] L. Wenyin & D. Dori. *Object-Process Based Graphics Recognition Class Library : Principles and Applications*. Software : Practice and Experience (SPE), vol. 15, no. 29, pages 1–24, 1999.
- [Winter 02] A. Winter, B. Kullbach & V. Riediger. *An Overview of the GXL Graph Exchange Language*. In International Seminar on Software Visualization, volume 2269 of *Lecture Notes in Computer Science (LNCS)*, pages 324–336, 2002.
- [Worboys 04] M. Worboys & M. Duckham. *Gis : A computing perspective*. CRC Press, 2 edition, ISBN : 0415283752, 2004.
- [Wu 93] W. Wu & M. Sakauchi. *A Multipurpose Drawing Understanding System with Flexible Object-Oriented Framework*. In International Conference on Document Analysis and Recognition (ICDAR), pages 870–873, 1993.
- [Xu 97] J.Z. Xu. *Rule-Based Graphics Recognition on Unconstrained Maps*. In Conference of Geospatial Information and Technology Association (GITA), 1997.
- [Yan 03] L. Yan & L. Wenyin. *Engineering Drawings Recognition Using a Case-based Approach*. In International Conference on Document Analysis and Recognition (ICDAR), pages 190–194, 2003.
- [Yan 04] L. Yan & L. Wenyin. *Interactive Recognizing Graphic Objects in Engineering Drawings*. In Workshop on Graphics Recognition (GREC), volume 3088 of *Lecture Notes in Computer Science (LNCS)*, pages 126–137, 2004.
- [Yaner 02] P.W. Yaner & A.K. Goel. *Retrieving 2-D Line Drawings by Example*. In Conference on Diagrammatic Representation and Inference (Diagrams), volume 2317 of *Lecture Notes in Computer Science (LNCS)*, pages 97–99, 2002.
- [Yu 97] Y. Yu, A. Samal. & S. Seth. *A System for Recognizing a Large Class of Engineering Drawing*. Pattern Analysis and Machine Intelligence (PAMI), vol. 19, no. 8, pages 868–890, 1997.
- [Zhang 04] D.Q. Zhang & S.F. Chang. *Stochastic Attributed Relational Graph Matching for Image Near-Duplicate Detection*. Rapport technique 206-2004-6, Department of Electrical Engineering, Columbia University, New York, USA, 2004.
- [Zou 00] J.J. Zou & H. Yan. *Vectorization of Cartoon Drawings*. In Workshop on Visual Information, pages 77–78, 2000.

# Table des figures

1.1	Système à Base de Connaissances (SBC)	1
1.2	Données, informations et connaissances	2
1.3	Formalismes logique/graphe	3
1.4	Exemple de différents modèles	3
1.5	Langages de représentation de graphes	4
1.6	Exemple de règle PROLOG	5
1.7	Exemple de graphes de primitives graphiques	6
1.8	Niveaux de représentation	8
1.9	Schéma relationnel des primitives graphiques	8
1.10	Graphes orienté/non orienté	9
1.11	Graphe sémantique/conceptuel	10
1.12	Exemple de schéma	10
1.13	Exemple de graphe relationnel attribué	12
1.14	Comparaison des modèles de [Han 94] et [Ramel 00]	16
1.15	Applications des modèles de [Han 94] et [Ramel 00]	17
1.16	Les différentes classes de modèles	18
1.17	1 algorithme vers $n$ représentations	21
1.18	$n$ algorithmes vers $n$ représentations	21
1.19	Problématiques multi-représentation vs interopérabilité	22

# Liste des tableaux

1.1 Représentations des connaissances graphiques (1) . . . . .	14
1.2 Représentations des connaissances graphiques (2) . . . . .	15

# Table des matières

<b>1</b>	<b>Gestion des connaissances graphiques, état de l'art</b>	<b>0</b>
1.1	Introduction . . . . .	0
1.2	Systèmes à base de connaissances . . . . .	0
1.2.1	Introduction . . . . .	0
1.2.2	Données, informations et connaissances . . . . .	1
1.2.3	Représentation, formalisme et modèle . . . . .	2
1.2.4	Langage, opérationnalisation et interopérabilité . . . . .	4
1.2.5	Conclusion . . . . .	6
1.3	Gestion des connaissances graphiques . . . . .	6
1.3.1	Introduction . . . . .	6
1.3.2	Les primitives graphiques . . . . .	8
1.3.3	Formalismes à base de graphes . . . . .	9
1.3.4	Représentations des connaissances graphiques . . . . .	12
1.3.5	Multi-représentation . . . . .	19
1.3.6	Conclusion . . . . .	22
1.4	Conclusion . . . . .	23
	<b>Bibliographie</b>	<b>25</b>
	<b>Table des figures</b>	<b>33</b>
	<b>Liste des tableaux</b>	<b>34</b>