



**HAL**  
open science

# Interprétation de Documents Techniques : des Outils à leur Intégration dans un Système à Base de Connaissances

Sébastien Adam

► **To cite this version:**

Sébastien Adam. Interprétation de Documents Techniques : des Outils à leur Intégration dans un Système à Base de Connaissances. Traitement du texte et du document. Université de Rouen, 2001. Français. NNT: . tel-00671320

**HAL Id: tel-00671320**

**<https://theses.hal.science/tel-00671320>**

Submitted on 17 Feb 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITE DE ROUEN-MONT SAINT AIGNAN  
U.F.R. DE SCIENCES ET TECHNIQUES

N° attribué à la thèse

THESE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE DE ROUEN

*Discipline : Sciences Appliquées*

Présentée et soutenue publiquement

Par **Sébastien ADAM**

Le 11 décembre 2001

Titre

**Interprétation de Documents Techniques :  
des Outils à leur Intégration dans un  
Système à Base de Connaissances**

*Directeur de thèse :*

Professeur Jacques Labiche

**COMPOSITION DU JURY**

Mme	Nicole	VINCENT	Université de Tours	Rapporteur
M.	Jean-Marc	CHASSERY	LIS, Grenoble	Rapporteur
M.	Joël	GARDES	France Télécom R&D	Examineur
M.	Jacques	LABICHE	Université de Rouen	Examineur
M.	Jean-Marc	OGIER	Université de La Rochelle	Examineur
M.	Karl	TOMBRE	LORIA, Nancy	Examineur



## Résumé

Les travaux présentés dans ce mémoire abordent la problématique de l'interprétation de documents techniques. Dans ce contexte, ils se trouvent à la confluence de différentes thématiques de recherche telles que le traitement du signal et des images, la reconnaissance de formes, l'intelligence artificielle, la communication Homme/Machine et l'ingénierie des connaissances. En effet, si ces domaines scientifiques diffèrent dans leurs fondements, ils sont complémentaires et leurs apports respectifs sont indispensables pour la conception d'un système d'interprétation fiable et adaptable. Dans ce contexte pluridisciplinaire, le mémoire est organisé en deux parties.

La première partie propose une méthodologie originale permettant la détection et la reconnaissance de formes (caractères et symboles) multi-orientées et multi-échelles. L'approche adoptée est basée sur la transformée de Fourier-Mellin. Elle permet la reconnaissance de formes isolées, mais aussi, dans une certaine mesure, de formes connectées. Son utilisation autorise en outre l'estimation des paramètres de mouvements des formes. Les outils développés sont évalués et comparés sur différentes bases de caractères et les résultats obtenus sont tout à fait compétitifs au regard des approches de la littérature.

La seconde partie de ce mémoire aborde quant à elle la problématique de l'interprétation de documents techniques avec un point de vue orienté vers l'ingénierie des connaissances. Les réflexions proposées dans ce cadre permettent selon nous de montrer la faisabilité et la pertinence d'une démarche orientée connaissances pour la conception d'un système d'interprétation. Elles ont donné lieu à une implémentation conduisant à un système nommé *NATALI v2*. Une représentation explicite des connaissances, une architecture logicielle à base d'agents ainsi que différentes interfaces homme-machine offrent une bonne adaptabilité et une grande souplesse au système.

**Mots clés** : analyse d'image, interprétation de documents techniques, reconnaissance de formes multi-orientées et multi-échelles, transformée de Fourier-Mellin, représentation de connaissances, système multi-agents, interfaces homme-machine

## Abstract

This thesis tackles the problem of technical document interpretation applied to France Telecom Documentation. This subject is on the crossroad of different fields like signal or image processing, pattern recognition, artificial intelligence, man-machine interaction and knowledge engineering. Indeed, each of these different fields can contribute to build a reliable and efficient document interpretation device. In this interdisciplinary context, this thesis is divided in two main parts.

The first part is considering an original method used to detect and recognise multi-scaled and multi-oriented patterns like symbols or characters. The theoretical basis of this method is given by the Fourier-Mellin transform. It allows recognising isolated patterns but also, in some cases, connected patterns. The approach also allows the estimation of shape's movement parameters. Tools that have been developed in this context are evaluated regarding the state of the art in optical characters recognition. Obtained results with this original method are really competitive.

The second part is focusing the theme of technical document analysis under the point of view of knowledge engineering. The aim is to show the feasibility and relevance of a "knowledge based approach" in the context of technical document interpretation. An external and explicit knowledge model, a distributed agent-based software architecture and several user interfaces give the main concepts of this approach. A first implementation using these concepts is shown through a presentation of a system named "NATALI v2". This implementation has good reliability and adaptability properties.

**Keywords** : digital image processing, technical document interpretation, multi-oriented and multi-scaled pattern recognition, Fourier-Mellin transform, knowledge representation, multi-agent systems, user interface.



---

# Remerciements

Ce travail de thèse a été réalisé dans le cadre d'une convention CIFRE au sein de deux organismes de recherche : le laboratoire **P**erception, **S**ystèmes et **I**nformation (**P.S.I.**) de l'université de Rouen et le département **G**estion et **R**echerche d'**I**nformations (**G.R.I.**) de la **D**irection des services **M**ultimédia **I**nternet et **I**ntranet (**D.M.I.**), de France Telecom R&D –Lannion.

Je tiens tout d'abord à adresser mes plus vifs remerciements aux personnes qui me font l'honneur de participer au jury de cette thèse. Dans ce cadre, je remercie tout particulièrement Madame Nicole Vincent, professeur à l'Université de Tours, chercheur au **L**aboratoire d'**I**nformatique (**L.I.**) de cette université, et Monsieur Jean-Marc Chassery, Directeur de Recherche CNRS au **L**aboratoire des **I**mages et des **S**ignaux (**L.I.S.**) de Grenoble pour l'intérêt qu'ils ont porté à ce travail en acceptant d'en être les rapporteurs. Je remercie également Monsieur Karl Tombre, Professeur à l'École de Mines de Nancy et chercheur au **L**aboratoire **L**Orrain de **R**echerche en **I**nformatique et ses **A**pplications (**L.O.R.I.A.**) d'avoir accepté d'être examinateur de ma thèse.

Parmi les autres membres du Jury, mes remerciements vont aussi à Monsieur Jacques Labiche, Professeur à l'Université de Rouen, et maintenant Directeur du P.S.I. Il a été mon directeur de thèse pendant les trois années passées. Je lui suis tout particulièrement reconnaissant d'avoir accepté cette responsabilité et d'avoir orienté mon travail avec dynamisme, discernement et une bonne humeur exemplaire.

Je remercie vivement Joël Gardes, qui participe à ce jury en tant que directeur industriel de cette thèse, même s'il déteste ce terme. Ses compétences et sa gentillesse légendaire ont permis que ces trois années se déroulent dans un excellent climat. Je pense que nos relations ont largement dépassé le côté professionnel, et j'espère avoir encore longtemps l'occasion de travailler avec lui.

Enfin, je terminerais les remerciements aux membres du jury par ceux adressés à Jean-Marc Ogier. Il a été l'encadrant principal de ce travail pendant trois ans. Il a su m'épauler et me guider à tous les moments par ses conseils et son enthousiasme hors pair, tout en me laissant beaucoup d'autonomie. Son influence sur mon travail n'est pas mesurable et je désire par ces quelques mots lui témoigner toutes ma reconnaissance et toute mon amitié. Le fait que je souhaite poursuivre le métier de chercheur lui incombe en grande partie.

Côté laboratoires d'accueil, je tiens à remercier Jean-Pierre Pécuchet pour son accueil au sein du laboratoire PSI. Mes remerciements vont aussi à mes responsables France Telecom, qui m'ont permis de bénéficier de très bonnes conditions de travail. Je tiens dans ce cadre à remercier Pierre-Yves Cochet, Marc Giraud, Luc Mathan et Thierry Stunault.

Dans l'équipe du projet, je tiens également à remercier Claude Cariou. Sans sa participation, la première partie de cette thèse n'existerait probablement pas. Les moments passés avec lui ont toujours

été très instructifs et extrêmement agréables, et ce que nos discussions abordent la TFM ou le chauffage à l'énergie solaire.

Côté P.S.I., là encore les personnes auxquelles je voudrais témoigner ma reconnaissance sont très nombreuses. Parmi celles-ci, mes « chefs » de la première heure : Rémy Mullot et Yves Lecourtier. Après avoir été mes enseignants, ils sont devenus des collègues pour lesquels j'ai beaucoup d'admiration et de sympathie. Je tiens à les remercier pour leur accueil dans l'équipe et pour les nombreuses discussions que nous avons eues. Mes remerciements vont aussi aux autres chercheurs du laboratoire, qu'ils soient thésards ou maîtres de conférence. Je les salue et les remercie tous, avec une mention particulière pour Ali, Pierre, Mathieu, Youssouf, Icham, Arnaud, Franck, Abdel, Laurent, Eric, Thierry, Gilles, Vincent et Patrick. Mes remerciements vont aussi à Nadia, Sandrine et Laurence que je harcèle en permanence.

Toute ma gratitude va aussi à Laurent qui a occupé une partie de son temps libre à la relecture de ce manuscrit.

Enfin, je terminerai ces remerciements par mes proches. Côté famille d'abord, je souhaite adresser tous mes remerciements à mes parents. Sans eux et les sacrifices qu'ils ont fait pour me permettre de suivre ces longues études, je n'en serais pas là. Qu'Hélène et Nico trouvent également ici l'expression tous mes remerciements. Ils sont un soutien de tous les instants et ils m'ont, entre autre, mis en condition chaque dimanche soir pour la semaine de labeur qui m'attendait, surtout ces derniers mois.

Côté amis, la liste est très longue, mais je veux vivement remercier tous ceux qui ont partagé mes soirées, week-end et vacances pendant ces trois ans. Parmi ceux-ci, un grand merci va à la famille Dudu, à Ninie, à Arnaud, à Ivanne, à Vincent et sa bande, à Manu, à Tit Fred, aux Anne's, à Laetitia, à la Gigi's band, à Georges, à Marie, à Agnès, à Cynthia, à Jack, à Paulette, et à tout ceux que j'oublie mais qui m'ont permis de passer de bons moments...

---

# Table des matières

1.	Introduction générale .....	1
2.	Des outils pour la reconnaissance de texte et de symboles .....	8
2.1.	INTRODUCTION : LA PROBLEMATIQUE.....	8
2.1.1.	<i>Les particularités du texte</i> .....	8
2.1.2.	<i>Les particularités des symboles</i> .....	11
2.1.3.	<i>Vers un modèle commun de reconnaissance</i> .....	12
2.1.4.	<i>Proposition de plan</i> .....	16
2.2.	LES MODELES DE REPRESENTATION DES FORMES DE LA LITTERATURE .....	17
2.2.1.	<i>Introduction : articulation de la synthèse et critères d'évaluation</i> .....	17
2.2.2.	<i>Les modèles issus de l'image en niveaux de gris</i> .....	18
2.2.3.	<i>Les modèles issus de l'image binaire</i> .....	29
2.2.4.	<i>Les modèles issus des contours de la forme</i> .....	37
2.2.5.	<i>Les modèles issus du squelette des formes</i> .....	42
2.2.6.	<i>Synthèse de l'étude bibliographique</i> .....	44
2.3.	UN NOUVEAU MODELE DE REPRESENTATION DES FORMES BASE SUR LA TRANSFORMEE DE FOURIER MELLIN (TFM).....	46
2.3.1.	<i>Quelques rappels</i> .....	46
2.3.2.	<i>Définition de la Transformée de Fourier Mellin (TFM)</i> .....	48
2.3.3.	<i>Des invariants issus de la transformée de Fourier Mellin</i> .....	49
2.3.4.	<i>Application à la structure bidimensionnelle discrète des images</i> .....	50
2.3.5.	<i>Estimation des paramètres de mouvement</i> .....	52
2.3.6.	<i>Algorithmes de mise en œuvre</i> .....	53
2.3.7.	<i>Résultats qualitatifs visuels</i> .....	58
2.3.8.	<i>Conclusion</i> .....	62
2.4.	EVALUATION QUANTITATIVE DES MODELES : LES RESULTATS OBTENUS.....	64
2.4.1.	<i>Introduction</i> .....	64
2.4.2.	<i>Les résultats obtenus pour les formes isolées</i> .....	64
2.4.3.	<i>Les résultats obtenus pour les formes connectées</i> .....	75
2.4.4.	<i>Les résultats pour l'estimation des paramètres de mouvement</i> .....	76
2.4.5.	<i>Conclusion et discussion sur l'évaluation</i> .....	79
2.5.	CONCLUSION.....	81
3.	Intégration des outils dans un système à base de connaissances.....	84
3.1.	INTRODUCTION .....	84
3.2.	LES CONNAISSANCES D'UN SYSTEME D'INTERPRETATION.....	87
3.2.1.	<i>Introduction</i> .....	87
3.2.2.	<i>Connaissances sur les données (connaissances descriptives)</i> .....	87
3.2.3.	<i>Connaissances sur les traitements (connaissances opératoires)</i> .....	90
3.2.4.	<i>Connaissances sur les stratégies (connaissances stratégiques)</i> .....	91
3.2.5.	<i>Conclusion</i> .....	92
3.3.	LES CONNAISSANCES DANS LA LITTERATURE DU DOMAINE .....	93
3.3.1.	<i>Dans les systèmes d'interprétation de documents techniques</i> .....	93
3.3.2.	<i>Dans les systèmes de vision en général</i> .....	110
3.3.3.	<i>Un bilan global des approches de la littérature</i> .....	123
3.4.	L'EXPERIENCE DE NATALI VI.....	125



3.4.1.	<i>Les objectifs</i> .....	125
3.4.2.	<i>Les connaissances et leur représentation</i> .....	126
3.4.3.	<i>L'implémentation de NATALI v1</i> .....	132
3.4.4.	<i>Analyse critique</i> .....	134
3.5.	DE L'ANALYSE CRITIQUE A NATALI V2.....	136
3.5.1.	<i>Introduction</i> .....	136
3.5.2.	<i>Les connaissances du domaine</i> .....	138
3.5.3.	<i>Les connaissances pour la reconnaissance des objets</i> .....	146
3.6.	CONCLUSION.....	156
4.	Conclusion générale et perspectives.....	159
	Bibliographie.....	165
	Annexe A. Principes des algorithmes d'optimisation utilisés.....	175
	A.1. PRINCIPES DE LA METHODE DU SIMPLEX.....	175
	A.2. PRINCIPES DES ALGORITHMES GENETIQUES (A.G.).....	176
	Annexe B .Un modèle statistico-structurel pour la reconnaissance de symboles.....	181
	B.1. LE PROBLEME.....	181
	B.2. L'APPROCHE CHOISIE.....	182
	B.3. LES RESULTATS OBTENUS.....	183
	Annexe C .Un cas d'usage de NATALI V2.....	187
	C.1. LE PROBLEME CONSIDERE.....	187
	C.1.1. <i>L'image traitée</i> .....	187
	C.1.2. <i>Les intentions d'interprétation</i> .....	188
	C.2. DEROULEMENT DE L'INTERPRETATION.....	188
	C.2.1. <i>La phase de propagation de l'intention</i> .....	189
	C.2.2. <i>La phase de remontée des informations</i> .....	190
	Les Figures.....	197
	Les Tableaux.....	198

---

# Introduction Générale

## 1. Introduction générale

### Le document comme support d'information

Avec l'essor grandissant des architectures de réseaux à travers le monde, le document électronique est devenu un vecteur incontournable pour la transmission d'idées et d'informations inter ou intra organisations. Cette émergence du document électronique a entraîné la naissance d'une nouvelle discipline : la **Gestion Electronique de Documents (G.E.D.)**. Elle consiste à assister les utilisateurs dans la production de leurs documents, dans la recherche rapide d'informations dans une base documentaire et dans la circulation des informations entre interlocuteurs. Les systèmes de **G.E.D.** sont désormais utilisés par la majorité des administrations, des entreprises et des autres organisations pour la capitalisation de leurs connaissances. La plupart des documents circulant sur les réseaux mondiaux en sont issus. Néanmoins, si aujourd'hui les documents sont dans leur majorité réalisés à l'aide d'outils informatiques et peuvent donc être directement intégrés dans des systèmes de **G.E.D.**, force est de constater que les documents qui peuvent être qualifiés de structurellement pauvres restent encore omniprésents. Les documents papier en sont la plus courante illustration. La mise à disposition de ceux-ci dans les circuits de l'information implique un processus plus ou moins automatisé de rétro conversion de l'image du document. Cette problématique de la conversion se retrouve également pour d'autres formats de représentation bas niveaux (PostScript, PDF, pages HTML à base de GIF) qui posent les mêmes types de problèmes lorsqu'il s'agit d'organiser l'information utile, de l'indexer, et de la retrouver aisément, afin de l'exploiter pour prendre des décisions. C'est dans ce contexte global de l'alimentation de systèmes de **G.E.D.** que se situent les travaux présentés dans cette thèse. En effet, si les processus de conversion ont atteint une relative maturité pour les documents structurés (formulaires, chèques, enveloppes...), la conversion des documents dits techniques, qui sont caractérisés par une association intrinsèque de graphique et de texte, est quant à elle encore du domaine de la recherche. Les documents techniques sont pourtant omniprésents dans les systèmes de **G.E.D.** et constituent souvent des supports décisionnels. Parmi la large variété de

documents techniques, citons, à titre d'exemple, les documents architecturaux, les schémas mécaniques, les cartes géographiques, les partitions de musique et les plans de réseau (Figure 1.1).

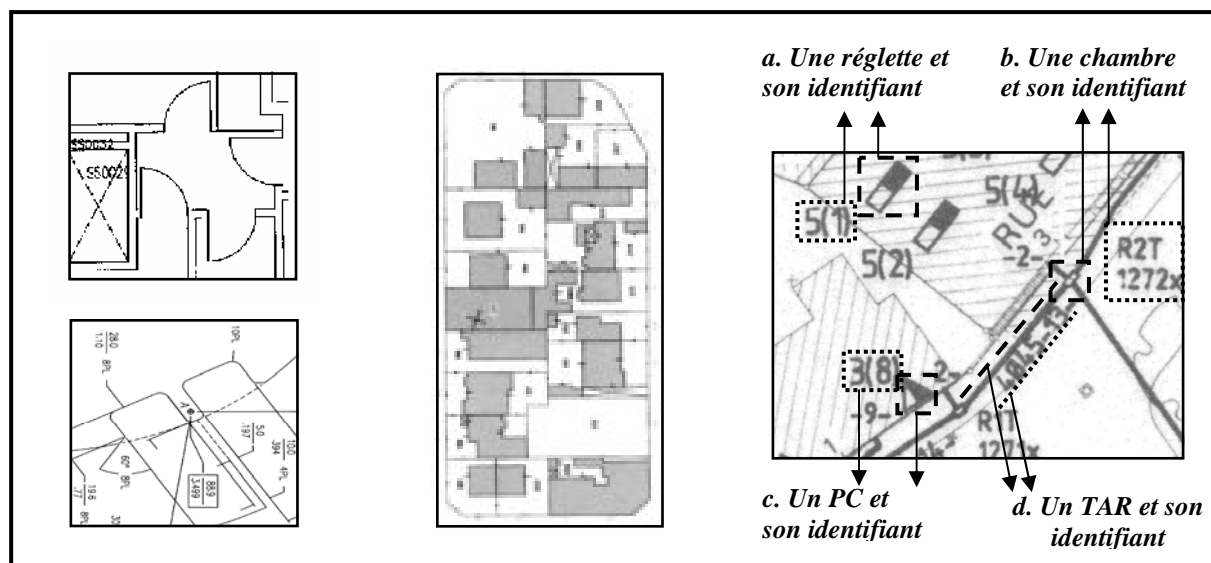


Figure 1.1 : Quelques exemples de documents techniques

## La conversion des documents techniques : un processus d'interprétation

Les différents types de documents évoqués ci-avant ont donné lieu à des études pour leur rétro conversion. Une telle conversion est un processus associant une acquisition –une numérisation dans le cas du papier, une fouille dans le cas de documents numériques–, suivie d'une indexation et/ou d'une interprétation. Cette seconde étape consiste à extraire du document toutes ou partie des informations du domaine que les concepteurs y ont fait figurer, en fonction des attentes de l'utilisateur du système. Un processus d'interprétation peut donc être assimilé à une tâche d'instanciation progressive d'un modèle issu du domaine du document ou d'une partie de ce modèle. Cette instanciation est réalisée en utilisant des procédures de traitement d'images et de reconnaissance de formes, elles-mêmes guidées par une stratégie d'interprétation qui doit être déduite d'une requête initiale et d'éventuelles coopérations avec l'utilisateur. Les disciplines scientifiques mises en jeu pour concevoir un système d'interprétation sont donc multiples : la reconnaissance de formes, l'intelligence artificielle ou la communication homme-machine en sont les illustrations les plus évidentes. La reconnaissance de formes vise à reconnaître le texte ou la représentation graphique d'entités du domaine présents sur l'image. L'intelligence artificielle aspire au développement de modèles pour l'extraction et la formalisation des connaissances et du raisonnement. La communication homme-machine permet quant à elle l'initialisation du système, sa supervision, ainsi que la présentation et la correction des résultats obtenus. C'est à la confluence de ces différents grands thèmes scientifiques que se trouve selon nous la solution au problème de l'indexation et de l'interprétation de documents techniques. Ces différentes thématiques seront donc abordées dans ce mémoire.

## Contexte industriel et objectifs de la thèse

Les préoccupations de l'opérateur de télécommunication France Telecom sont une parfaite illustration des considérations évoquées ci-avant, tant du point de vue de la rétro conversion de documents papiers pour l'alimentation du système d'information, que du point de vue de l'indexation de documents. Il existe en effet des masses considérables de documents techniques au sein de l'entreprise. C'est pourquoi le centre de recherche de France Telecom (le CNET<sup>1</sup>, puis France Telecom R&D<sup>2</sup>) s'est intéressé, depuis 1997, au problème de la conversion de documents techniques. C'est dans ce contexte qu'un partenariat a été établi entre notre équipe, le laboratoire LASTI<sup>3</sup> de l'ENSSAT<sup>4</sup> de Lannion, et France Telecom R&D. Ce partenariat a donné lieu à deux thèses : celle de Vincent Grenier [Grenier 2001], soutenue en juin 2001 et celle correspondant à ce manuscrit. Les objectifs initiaux de ces travaux, définis dans le cadre d'un projet nommé *NATALI*<sup>5</sup>, étaient de développer un système d'interprétation de plans d'itinéraires, afin d'alimenter la montée en charge de l'application de gestion du réseau France Telecom.

Plus récemment, l'orientation donnée à l'entreprise a pris une forte coloration fournisseur de services Internet. D'une logique de rétro conversion exhaustive d'un corpus de documents patrimoniaux, nous sommes donc passé à une logique de caractérisation et d'indexation de contenus de documents hétérogènes (multi-sources, multi-formats). Dans ce contexte, un nouveau projet nommé *DocMining* a vu le jour. Son objectif est de se placer dans une logique de *back office* pour concevoir puis organiser la distribution d'informations urbaines de proximité portées par un corpus de documents fortement hétérogènes. Le principe est d'inclure les couches de service appropriées pour exploiter les contenus de ces documents. Le premier prototype vise à utiliser de la documentation allant du plan de masse aux « simples » prospectus, pour fournir des services de type « plan d'accès ». L'extraction de ces couches de services exploitera des méthodes de reconnaissance de formes et sera incluse dans un système piloté par des bases de connaissances. De plus, ce système ne pouvant être généraliste du fait de la diversité des informations à reconstruire, les interfaces homme-machine doivent permettre d'initialiser le système et d'en assurer la supervision.

C'est dans ce contexte que la présente thèse a été définie. Les problématiques évoquées ci-dessus y sont abordées, et resituées dans des problèmes plus généraux d'analyse et d'interprétation d'images. Si les différents concepts et implémentations retenus concernent avant tout le projet initial *NATALI*, un effort a été consenti pour que les choix effectués restent adaptables pour le projet *DocMining*.

---

<sup>1</sup> Centre National d'Etude des Télécommunications

<sup>2</sup> France Telecom Recherche et Développement, Lannion

<sup>3</sup> Laboratoire d'Analyse des Systèmes de Traitement de l'Information

<sup>4</sup> Ecole Nationale Supérieure de Sciences Appliquées et de Technologie

<sup>5</sup> Numérisation et Acquisition pour le Traitement Automatique et la Lecture de l'Itinéraire des réseaux

## **Plan de la thèse**

Cette thèse s'articule autour de deux grandes parties thématiques, elles-mêmes décomposées en chapitres. Ces deux parties correspondent aux deux domaines scientifiques majeurs abordés au cours de ce travail : la reconnaissance de formes et l'intelligence artificielle.

La première partie s'intitule « Des outils pour la reconnaissance de texte et de symboles ». Elle se focalise sur l'un des problèmes fondamentaux de l'analyse de documents : l'extraction du texte et des symboles. Ce problème revêt en effet une importance considérable due à la richesse sémantique du texte et des symboles sur des documents techniques. Un chapitre introductif débute cette partie en présentant les différentes contraintes auxquelles est confronté un tel processus de reconnaissance. Un modèle de stratégie visant à résoudre ce problème est alors proposé. Il est basé sur deux étapes principales : l'extraction d'un modèle de représentation des formes et un processus d'appariement de modèles. Le second chapitre se focalise sur la première étape, en dressant une liste catégorisée des modèles de représentation qui sont retenus dans la littérature pour décrire des formes. Le chapitre 3 décrit le choix retenu par notre équipe, qui est basé sur l'utilisation de la transformée de Fourier Mellin pour décrire les formes indépendamment de leur position, de leur échelle et de leur orientation. Le quatrième chapitre expose les résultats obtenus par l'utilisation de ce modèle pour la reconnaissance des caractères isolés sur différentes bases, mais aussi pour l'extraction de caractères collés et pour l'estimation de l'orientation de ces caractères. Cette partie se conclut par une discussion sur la méthodologie d'évaluation adoptée, et par une réflexion sur les travaux menés.

La seconde partie s'attache à des problèmes de plus « haut niveau », en abordant des aspects liés à l'ingénierie des connaissances et aux interfaces de dialogue, afin de prendre en compte l'hétérogénéité des supports. Cette partie s'intitule « intégration des outils dans un système à base de connaissances ». Après un premier chapitre introductif présentant notre démarche, un second chapitre dresse un bilan catégorisé des connaissances qui sont manipulées par un système d'interprétation de documents techniques. Le chapitre 3 décrit les choix effectués dans la littérature pour modéliser, représenter et opérationnaliser ces connaissances, en suivant la catégorisation proposée dans le chapitre précédent. Le quatrième chapitre décrit le premier système conçu et développé par notre équipe, dans le cadre du projet *NATALI*, en prenant le même point de vue que pour l'analyse de la littérature et en en dressant une analyse critique. Puis, dans le chapitre 5, nous proposons quelques améliorations de ce système du point de vue de l'acquisition, de la représentation et de l'évolution des connaissances du système. Certaines de ces propositions ont donné lieu à des implémentations alors que d'autres restent au niveau conceptuel.

Le mémoire se termine par une synthèse des travaux réalisés dans le cadre de cette thèse. Les points faibles et les points forts sont soulignés. Cette critique du travail permet alors d'exposer les perspectives envisageables.

Des annexes abordent quelques points particuliers : la présentation de deux algorithmes d'optimisation et celle d'un système de reconnaissance de symboles développé au cours du DEA qui a précédé cette thèse et qui a donné lieu à des travaux ultérieurs.



---

# **Partie II. Des Outils pour la Reconnaissance de Texte et de Symboles**

- 2.1. Introduction : la problématique**
- 2.2. Les modèles de représentation des formes dans la littérature**
- 2.3. Un nouveau modèle de représentation des formes basé sur la transformée de Fourier Mellin**
- 2.4. Evaluation quantitative des modèles : les résultats obtenus**
- 2.5. Conclusion**



## **2. Des outils pour la reconnaissance de texte et de symboles**

### **2.1. Introduction : la problématique**

Dans le cadre de l'introduction générale de ce mémoire, nous avons mis en avant le fait que la reconnaissance du texte et des symboles constituait une étape prépondérante de l'interprétation d'un document technique. En effet, quel que soit le document technique considéré, celui-ci est porteur d'une symbologie particulière, en relation avec le métier concerné. Cette symbologie est fondamentale pour l'expert du domaine qui exploite les documents. Ainsi, sur un plan architectural par exemple, les portes et les fenêtres sont représentées par des symboles. Sur une carte géographique, un lecteur humain reconnaît rapidement des hôpitaux ou un syndicat d'initiative, grâce aux éléments symboliques qui leur sont propres. Sur un schéma mécanique, les symboles tels que ceux représentant des engrenages permettent au spécialiste de comprendre la cinématique associée à la pièce. Sur une partition, le musicien reconnaît les notes, les clefs, les bémols, les dièses... Sur un schéma électronique, ce sont les composants qui sont représentés par l'intermédiaire des symboles. Enfin, sur un document de réseau (gaz, téléphone, électricité), les symboles représentent la présence d'infrastructures nécessaires à la gestion du réseau.

Pour chacun de ces exemples illustrant l'importance des symboles, un complément d'information est souvent apporté sur le document sous forme de texte écrit à proximité du symbole. La sémantique portée par cette information textuelle est variable suivant la nature des documents (thématiques, fonctionnels, informationnels). Néanmoins, comme la reconnaissance des symboles, la lecture automatique de l'information textuelle est un maillon indispensable à tout système d'interprétation, pour la production de données cohérentes. Cette partie de la thèse est dédiée à ce problème de la reconnaissance du texte et des symboles, en se plaçant au « bas niveau ». Par bas niveau nous signifions ici que les informations du domaine telles que la correspondance texte / symbole ou un éventuel lexique par exemple ne sont pas exploitées. Seuls les pixels constituant la forme sont pris en compte. Nous traitons donc ici de ce que certains auteurs tels que Karl Tombre ou Rémy Mullot appellent « extraction d'indices ». De ce fait, tout ce qui est présenté dans cette partie de la thèse est générique et parfaitement exploitable pour l'analyse de tout type de document technique. L'intégration de ces outils dans un système à base de connaissances sera quant à elle au centre de la seconde partie de cette thèse.

#### **2.1.1. Les particularités du texte**

Afin de comprendre les contraintes auxquelles est soumis un système d'interprétation de documents techniques, en ce qui concerne la reconnaissance des informations textuelles, nous présentons ci-dessous l'ensemble des catégories de texte qui peuvent être rencontrées sur de tels documents. Nous

illustrons nos propos par quelques exemples tirés des différentes catégories de documents techniques évoquées dans l'introduction générale.

- Un éventuel « titre » du document. Dans le cas d'une partition musicale, il s'agit du morceau composé. Sur une carte urbaine ou un plan de réseau, c'est le nom de la ville représentée qui fera office de titre ;
- Des informations générales sur le document, souvent sous la forme de « cartouche ». Dans le cas de cartes ou de plans de réseau, cela inclut des informations sur le géoréférencement (souvent les coordonnées Lambert) qui permettent de reconstituer le continuum géographique du territoire avec l'ensemble des plans d'une zone. Des informations sur l'historique du document, avec les dates de création ou de mise à jour du plan par exemple sont également des informations fréquemment retrouvées sur des documents techniques ;
- Des identifiants propres aux objets du domaine présents sur le document. Pour un schéma électronique par exemple, les composants sont souvent numérotés dans un ordre donné ( $R_1, R_2 \dots R_N$  pour les différentes résistances). Sur un plan architectural, les noms des pièces (Chambre, cuisine...) apparaissent dans les zones correspondantes. Sur un plan cadastral, ce sont les numéros de parcelles qui identifient les objets. Sur les plans de réseau France Telecom, une chambre sera identifiée par la présence d'une chaîne « L2T » ou « R2T » (b. sur la Figure 1.1) ;
- Du texte définissant des propriétés associées aux objets du domaine. Ces informations permettent de caractériser les objets de façon plus précise, en fournissant des informations propres à leur fonctionnalité ou à leurs dimensions par exemple. Dans le cas des plans architecturaux ou de schémas mécaniques, il peut s'agir d'informations de cotations qui doivent être en adéquation avec la représentation graphique de l'objet métier. Sur les plans de réseau, les représentations des câbles sont souvent associées à une chaîne de caractères qui donne leur longueur sur le terrain (d. sur la Figure 1.1) ;
- Une légende peut aussi être présente. Elle permet d'expliquer certaines représentations graphiques ; elle est toujours présente sur les cartes routières par exemple.

Quelle que soit la catégorie d'information textuelle évoquée ci-dessus, le texte constitue une information prépondérante pour la reconstruction sémantique des objets du domaine, puisqu'il sert de lien entre la représentation topologique (le plan) et les propriétés des objets du domaine. Sa reconnaissance est donc un maillon incontournable d'un système d'interprétation fiable. Or, dans le cadre de notre étude, comme pour la plupart des documents techniques, cette reconnaissance est soumise à un certain nombre de contraintes propres à ce type de documents, qui sont plus rarement présentes dans des documents structurés. Ces contraintes rendent bien souvent inutilisables les systèmes de reconnaissance commerciaux classiques. Dans le cadre de notre étude, ces particularités à

prendre en considération sont les suivantes (une illustration de ces contraintes dans le cas de plans de réseau France Telecom est donnée en Figure 2.1).

- Les caractères peuvent apparaître soit regroupés en mots, soit isolés ;
- Le nombre de fontes (nous utilisons ce terme même s'il ne s'agit pas réellement d'une fonte) est très restreint, la plupart du temps réduit à un, puisque le dessinateur utilise généralement un normographe pour écrire le texte sur les documents papiers. Néanmoins, quelques occurrences de texte manuscrit peuvent apparaître sur les documents lorsque des corrections ont été apportées au document original. Ce type de correction est souvent appelé annotations dans la littérature du domaine ;
- Les caractères peuvent être de tailles différentes. Ces différentes tailles permettent de faire la différence entre certains types d'informations. Par exemple, le nom des villes est bien souvent écrit dans une taille au moins deux fois plus grande que celle caractérisant un *PC*, la légende, quant à elle, est écrite dans une taille très petite...
- Les caractères sont représentés suivant différentes orientations. Si, dans le cas d'informations complémentaires apportées à des symboles, les toponymes sont dans la majeure partie des cas horizontaux ou verticaux pour en faciliter la lecture, lorsqu'il s'agit de caractériser des objets linéaires (comme les TAR ou les routes par exemple), l'axe de lecture du texte suit souvent le trait et peut donc admettre une orientation variant de  $-180$  à  $+180$  degrés. Cela peut également être le cas lorsqu'une zone d'un document est surchargée, et que le dessinateur n'a pu trouver la place d'écrire horizontalement. Néanmoins, force est de constater que le texte est rarement écrit à l'envers, pour des commodités de lecture des plans ;
- Les caractères sont parfois connectés à des éléments graphiques du plan. Il est très fréquent que sur l'image binaire issue de la numérisation de la planche (ou d'une numérisation en niveaux de gris suivie d'une binarisation mal paramétrée), des caractères se trouvent connectés au tracé des câbles ou à des symboles. Cela peut être dû à des erreurs du dessinateur, à un manque de place sur le plan papier ou encore à cause de problèmes d'acquisition. Néanmoins, quelle que soit la source, ce point particulier constitue un verrou méthodologique important que peu de systèmes sont en mesure de prendre en compte ;
- Les caractères sont parfois connectés entre eux. Pour les mêmes raisons que dans le cas d'une connexion au graphique, il est fréquent que deux caractères forment une seule composante connexe à l'issue de la numérisation. Là encore, les propositions pour résoudre ce genre de problèmes sont rares dans la littérature.

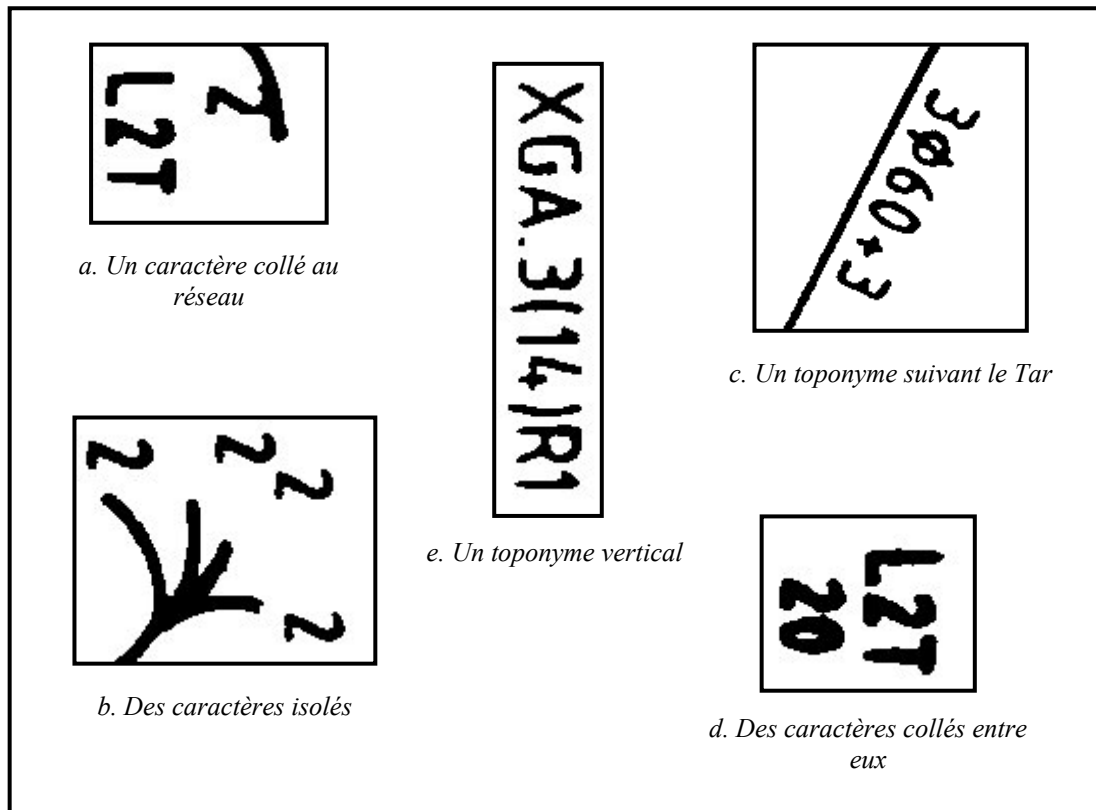


Figure 2.1 : Illustration des contraintes imposées au système de reconnaissance.

### 2.1.2. Les particularités des symboles

Sur un document technique (qu'il soit cartographique, mécanique, architectural, électrique...), les symboles constituent une source d'information importante puisqu'ils représentent des entités spécifiques du domaine. Ces données symboliques sont soumises aux mêmes contraintes que celles évoquées dans le cas des données textuelles, c'est-à-dire des orientations et échelles diverses, et des connexions possibles entre celles-ci et d'autres entités. Dans le cas des documents réseau France Telecom, les symboles représentent l'ensemble des infrastructures techniques qui, avec les câbles, permettent à l'opérateur d'offrir ses services téléphoniques ; leur représentation est liée à une légende qui est implicite puisque non reproduite sur l'ensemble des plans mais définie sur un document particulier unique, le **Cahier des Clauses Techniques Particulières (CCTP 1592)** qui regroupe l'ensemble des éléments devant composer un plan d'itinéraire. La Figure 2.2. propose un extrait de cette légende commune aux plans de réseau.

Si les données textuelles et les données symboliques ne sont pas de même nature d'un point de vue métier, nous allons maintenant voir que ces deux types de données peuvent en revanche être appréhendées dans un même modèle générique de reconnaissance lorsqu'ils sont isolés.












Type	Contenance	Symbole
Point de concentration sur appui de FRANCE TELECOM	3 ou 4 paires	
	7 paires	
	14 paires	
Point de concentration sur support d'énergie électrique	3 ou 4 paires	
	7 paires	
Point de concentration sur façade	3 ou 4 paires	
	7 paires	
	14 paires	
Point de concentration sur potelet	3 ou 4 paires	
	7 paires	
	14 paires	

Figure 2.2 : Un extrait du CCTP 1592, décrivant l'ensemble de la symbologie France Telecom

### 2.1.3. Vers un modèle commun de reconnaissance

Qu'il s'agisse du texte ou des symboles, le schéma de la reconnaissance au « niveau image » est relativement classique. Il consiste en l'enchaînement de trois étapes, parfois associées à une quatrième dédiée au traitement des formes collées (qui sont, nous l'avons dit, assez fréquentes sur nos documents). Nous présentons par la suite les trois étapes classiques avant d'aborder le problème des formes connectées :

- la localisation : la localisation des caractères et des symboles constitue la première étape d'un processus de reconnaissance. Elle consiste à isoler les caractères ou les symboles sur une couche particulière d'images qui sera soumise à la reconnaissance. Contrairement au cas d'un document structuré, il n'y a pas de notion de lignes de textes sur un document technique. De ce fait, les techniques classiques des systèmes commerciaux de reconnaissance, souvent basées sur la détection de lignes de texte par projection, deviennent inefficaces. Ce point particulier a donc donné lieu à la publication de nombreux travaux propres au domaine du document technique. En ce qui concerne le texte, les approches classiques sont basées sur une exploitation de l'information potentielle caractère, s'appuyant sur la constatation intuitive de Z. Lu : « la taille des caractères est

souvent plus petite que les éléments graphiques, et les caractères apparaissent souvent sous forme d'une chaîne dont l'orientation est horizontale, verticale ou oblique » [Lu 1998]. Notons que si l'information taille est effectivement exploitable dans tous les cas, le second point concernant le regroupement en chaînes peut, quant à lui, paraître discutable, en particulier sur les documents de notre étude puisque les caractères isolés sont fréquents. C'est pourquoi, comme [Trier 1997, Fletcher 1988, Lu 1998, Shimotsuji 1992, Lai 1994, Devaux 1999, Langrana 1997], nous avons appuyé notre localisation sur un filtrage des composantes connexes par la taille, en exploitant les informations largeur et hauteur du rectangle englobant, ainsi que le nombre de pixels noirs dans ce rectangle. Les seuils utilisés sont issus d'un apprentissage simple des limites inférieures et supérieures des différentes caractéristiques. Notons que s'il n'est pas exploité pour localiser les entités caractères, le chaînage de ceux-ci en toponymes est tout de même exploité afin de reconstruire des chaînes en adéquation avec le modèle du document. Ce point constitue la troisième étape qui sera décrite ci-après.

- La reconnaissance : si un fonctionnement parfait de l'étape précédente est admis (ce qui n'est pas nécessairement le cas, comme nous le verrons ultérieurement), à son issue, une image ne contenant que des caractères et/ou des symboles isolés les uns des autres est alors disponible. L'étape suivante consiste alors à reconnaître chacun d'eux, c'est à dire à émettre une hypothèse (voire une liste d'hypothèses) quant à leur classe. Quelle que soit l'approche adoptée, la reconnaissance d'une forme inconnue passe par une comparaison de celle-ci avec les éléments d'une base contenant des formes préalablement étiquetées. Pour ce faire, deux alternatives sont possibles :
  - Le *matching* d'image : une telle approche consiste à comparer directement l'image inconnue à une base d'images préalablement étiquetées, en utilisant une distance parmi les différentes distances entre images ayant été développées dans la littérature [Di Gesù 1999][Klette 1987]. Si cette approche est très fréquemment utilisée dans le cas de systèmes d'OCR dédiés à la reconnaissance de documents structurés, en s'affranchissant d'éventuels problèmes de taille par une normalisation, elle devient problématique lorsque la contrainte de multi-orientation intervient. En effet, dans ce cas, soit la distance adoptée est elle-même invariante au changement d'orientation, soit le système doit avoir dans sa base d'apprentissage l'ensemble des orientations possibles des formes. Pour chaque forme, chacune des orientations possibles peut alors en quelque sorte être assimilée à une nouvelle classe. Du fait du grand nombre de prototypes d'apprentissage alors nécessaires pour pallier la « malédiction de la dimensionnalité » (cf. 2.4.5), et de la complexité engendrée, peu d'équipes ont adopté une telle approche. Les contributions de Yang [Yang 1987], de Rummelhart [Rummelhart 1986], qui, à l'aide de réseaux de neurones multi-couches ont exploité une telle approche méritent néanmoins d'être citées dans ce cadre. Les travaux de

l'équipe de Yann Le Cun [Le Cun 1999] [Simard 2000] avec le réseau de neurones à convolution LeNet-5 entrent également dans ce cadre. Néanmoins, la tolérance à la multi-orientation est limitée dans ce travail à une variation de 40° des formes.

- Le passage par un changement de l'espace de représentation de la forme : pour pallier les difficultés évoquées ci-dessus, la reconnaissance de formes passe souvent par d'autres espaces de représentation dérivant de l'image initiale. Ces représentations ont pour but de synthétiser l'ensemble de l'information contenue dans la forme. De telles approches peuvent être assimilées à une compression dont le but n'est pas d'avoir le moins de perte possible mais le plus grand facteur discriminant. Par exemple, il est possible de convertir une forme en un graphe de vecteurs, un graphe d'occlusions, un vecteur statistique de caractéristiques afin de réduire « l'espace de représentation » pour faciliter la décision. Selon la vue adoptée, la reconnaissance de la forme passera alors par une comparaison à une base de vecteurs statistiques, un graphe modèle... Une telle approche, peut être illustrée par le modèle de reconnaissance d'une forme que nous présentons sous la forme d'un arbre des tâches à effectuer en Figure 2.3. Son parcours se fait en profondeur d'abord, puis de gauche à droite. Notons que dans la suite de ce mémoire, nous emploierons indifféremment les expressions « espace de représentation », « modèles de représentations » ou encore simplement « modèle » lorsque cela ne portera pas à confusion, pour nommer le résultat du changement de représentation des formes.

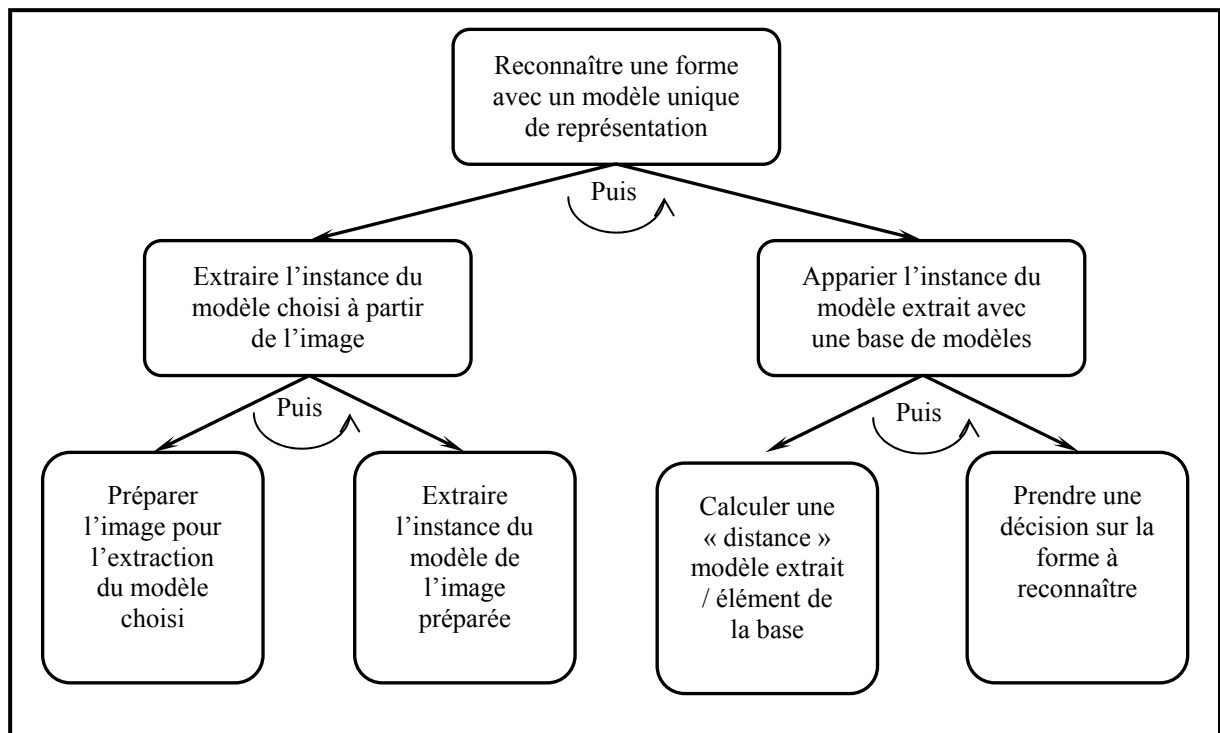


Figure 2.3 : Un Modèle générique de la reconnaissance d'une forme isolée utilisant un modèle unique

- Le regroupement en chaînes : cette étape n'est pas prise en compte pour la détection des caractères sur le plan, à cause de la fréquente apparition de caractères isolés. Néanmoins, cette information est tout de même ensuite exploitée pour reconstruire le sens du document. Dans ce domaine, la littérature est également assez abondante. Un algorithme très classique, et ayant fait ses preuves (dans [Dosch 2000] par exemple), est celui de Fletcher [Fletcher 1988]. Cette approche propose de déterminer les chaînes de caractères en se basant sur un ensemble d'heuristiques et en utilisant la Transformée de Hough pour déterminer les alignements. L'algorithme requiert 3 composantes connexes dont les centres sont alignés pour valider une zone « toponyme ». Il est à noter qu'une telle approche peut facilement se laisser piéger par la présence de lignes pointillées. Notons ici aussi les contributions plus récentes de Lu [Lu 1998] et Shimotsuji [Shimotsuji 1992], que nous ne détaillerons pas puisque, si leur mise en œuvre est différente, elles exploitent également les deux informations disponibles que sont colinéarité et proximité. Enfin, Deseilligny [Deseilligny 1995a, Deseilligny 1995b] propose une méthodologie de reconstitution de chaînes basée sur un algorithme de programmation dynamique permettant de construire le chaînage optimal en intégrant des contraintes de cohérence d'orientation des chaînes, la phase de reconnaissance des caractères étant menée au préalable.

Cette reconstitution de chaînes de caractères ne constitue pas l'objectif de cette thèse. Néanmoins, le développement de l'approche système présentée dans la seconde partie de cette thèse passait par l'utilisation d'un constructeur de toponymes. Nous avons donc repris les principes de l'algorithme de Fletcher en l'appliquant plus précisément à notre problème. Les développements et résultats obtenus sont détaillés dans [Lematre 1999].

- Le traitement des données (textuelles et symboliques) connectées : que ce soit dans le cas du texte ou dans le cas des symboles, nous avons évoqué précédemment le fait que ces données pouvaient être connectées entre elles ou à un autre élément du document. De telles connections sont très problématiques pour l'interprétation des documents. En effet, elles rendent impossible la détection des formes par des techniques classiques basées sur la taille des composantes connexes. De ce fait, le processus de conversion en une autre représentation pour sa reconnaissance n'est plus possible et les formes seront « oubliées ». De plus, si aucun traitement spécifique n'est mis en œuvre, ces formes connectées perturberont également le résultat du traitement des parties graphiques. Sur ce point, la littérature aborde essentiellement les méthodes ayant pour but d'éviter que le problème survienne. Les algorithmes publiés dans ce domaine sont généralement basés sur des opérateurs morphologiques [Lu 1998, Yu 2001] ou sur des segmentations de niveaux de gris adaptatives [Trier 1995a, Trier 1995b]. Le lecteur intéressé pourra trouver une bonne synthèse des méthodes de segmentation dans [Casey 1996]. Dans le cas où de telles méthodes échouent, la littérature apporte alors beaucoup moins de solutions. Le problème de la reconnaissance de telles données



peut alors être décomposé en deux sous-problèmes. Dans un premier temps, il s'agit de détecter la présence d'une telle connexion afin de pouvoir la traiter. Sur ce point particulier, les cas des caractères et des symboles peuvent être distingués. En effet, si dans le cas de caractères connectés (que ce soit à du graphique ou à d'autres caractères) l'information chaîne de caractères peut être exploitée [Deseilligny 1994], ce n'est pas le cas pour les symboles qui sont dans la majeure partie des cas isolés. La seconde étape consiste ensuite à reconnaître la forme dans la zone détectée.

#### **2.1.4. Proposition de plan**

C'est dans le cadre de la seconde étape d'un processus de reconnaissance du texte (reconnaissance des caractères eux-mêmes) que nous tentons d'apporter une contribution dans cette partie du mémoire. Plus précisément, dans la suite de cette partie dédiée à l'aspect « traitement du signal et des images », nous focalisons notre attention sur les modèles de formes qui sont utilisés pour la reconnaissance. La suite de cette partie s'articulera donc de la façon suivante ; dans un premier temps, nous dresserons un état de l'art des modèles développés pour la reconnaissance de formes dans la littérature. Cette synthèse montrera les avantages et inconvénients de chacune des approches au regard de notre étude. Puis, nous présenterons le modèle de formes que nous avons développé, basé sur la transformée de Fourier Mellin. Nous en présenterons à la fois les aspects théoriques et la mise en œuvre. Enfin, la dernière partie sera dédiée à l'évaluation de notre modèle, en comparant ce dernier aux modèles reconnus fiables dans la littérature.

## 2.2. Les modèles de représentation des formes de la littérature

### 2.2.1. Introduction : articulation de la synthèse et critères d'évaluation

Cette section dresse une synthèse de l'état de l'art concernant les techniques de représentation des formes pour leur reconnaissance. La littérature étant extrêmement abondante, nous avons voulu trouver une catégorisation des différentes approches. Dans ce contexte, plusieurs alternatives d'articulation s'offraient à nous. Citons par exemple la possibilité de catégoriser les approches en fonction de leur aspect local ou global (comme dans [Adam 2001a]) ou de leur aspect statistique ou structurel/syntaxique (SPR<sup>6</sup> versus SSPR<sup>7</sup>). Nous avons décidé de suivre la catégorisation proposée dans [Trier 1996] qui consiste à présenter les différentes approches en fonction de l'information qui est directement exploitée pour construire le modèle (image en niveaux de gris, image binaire, contour ou squelette). Ceci a pour but de permettre à une personne souhaitant construire une application de reconnaissance de formes de choisir en fonction des données dont elle dispose. Outre ce choix d'articulation, un autre problème qui se posait pour la présentation d'une telle synthèse était d'avoir un point de vue critique sur les méthodes exposées. En effet, la majeure partie des auteurs présente l'application de leur modèle et les résultats que celui-ci permet d'obtenir pour un problème particulier. D'un article à l'autre les bases d'apprentissage, les bases de tests ainsi que la méthodologie de prise de décision diffèrent donc considérablement. De plus, même si un modèle est parfaitement adapté à un problème de reconnaissance donné, il peut complètement échouer sur une autre problématique où les contraintes de reconnaissance sont différentes. Ce thème de l'estimation des performances n'est d'ailleurs pas propre au problème de la reconnaissance de caractères. De très nombreuses équipes travaillent actuellement à la mise en place de protocoles rigoureux d'évaluation de tous les types de traitements (vectorisation, binarisation...). Les travaux les plus encourageants dans ce domaine sont ceux réalisés pour la constitution de la base NIST, pour la reconnaissance de formulaires, de caractères manuscrits ou encore d'empreintes digitales. Quelques récentes manifestations, telles que les concours organisés lors des derniers *workshops* Graphic RECOgnition en 1999, témoignent de la prise en compte de cette problématique dans notre communauté. Néanmoins, à notre connaissance il n'existe pas encore de bases de comparaison pour les caractères multi-orientés et multi-échelles.

Toutes ces raisons expliquent pourquoi, dans le cadre de cette synthèse, nous tenterons essentiellement de mettre en avant les méthodes respectant du point de vue théorique les contraintes imposées à notre système. Nous accorderons davantage de précision aux méthodes que nous comparerons ultérieurement à la nôtre, à savoir les moments de Zernike [Teague 1980] et les sondes circulaires

---

<sup>6</sup> **SPR** : Statistical Pattern Recognition

<sup>7</sup> **SSPR** : Syntactical and Structural Pattern Recognition

[Lefrere 1993]. Ces outils ont été choisis, d'une part, pour la haute considération qui leur est accordée dans la littérature, et d'autre part pour leur complémentarité locale (Sondes) / globale (Zernike).

### 2.2.2. Les modèles issus de l'image en niveaux de gris

Dans cette partie, nous dressons une synthèse des travaux de la littérature concernant les modèles de représentation de formes qui peuvent théoriquement être extraits directement d'une image en niveaux de gris, sans utilisation de binarisation dans la chaîne de traitement. Nous présentons d'abord chacun des modèles d'un point de vue théorique. Puis nous dressons une synthèse de ces approches en 2.2.6.

#### 2.2.2.1. L'image et ses représentations dérivées

Dans le domaine du traitement d'images, les techniques de *template matching* sont souvent utilisées. Elles consistent à directement exploiter les valeurs des pixels (le *Bitmap*) en tant que modèle de la forme. Une normalisation de la taille de l'imagette contenant la forme peut être effectuée pour obtenir l'invariance aux changements d'échelle. Puis, une distance entre images (Hausdorff, Baddeley, Kulback...) est utilisée pour prendre la décision sur la classe de la forme. Un bon état de l'art sur les distances entre images en niveaux de gris est disponible dans [Di Gesù 1999]. Malgré leur apparente simplicité, de telles approches souffrent de nombreuses limitations. D'abord, il est nécessaire, pour obtenir l'invariance à la rotation, d'avoir des *templates* dans toutes les orientations possibles des formes. Il n'existe en effet pas, à notre connaissance, de distance entre images en niveaux de gris invariante à la rotation. De plus, ces approches se révèlent très sensibles aux bruits dans l'image ainsi qu'à de petites variations entre caractères d'une même classe, même dans le cas d'une orientation connue. Enfin, si la complexité d'extraction du modèle est réduite à celle de la normalisation de la taille de l'image de la forme, celle de la prise de décision est rendue importante par la grande taille de la base de *templates* nécessaire. Ces raisons expliquent pourquoi il n'existe à notre connaissance aucun système de reconnaissance de caractères multi-orientés s'appuyant sur de telles approches.

Sans utiliser l'image en niveaux de gris dans son intégralité, il est aussi envisageable, pour reconnaître une forme, d'exploiter un découpage de celle-ci en zones, et de calculer dans ces zones des valeurs statistiques qui synthétisent l'information. Là encore, l'invariance à la rotation semble difficile à obtenir si le découpage en zones n'est pas basé sur des cercles, ce qui ne semble pas avoir été exploité.

#### 2.2.2.2. Les transformées unitaires

Les Transformations Unitaires (TU) discrètes sont couramment utilisées pour le traitement et l'analyse d'images. Ces transformées permettent, par exemple, des applications de filtrage, de compression, de description d'images, tant pour des approches déterministes que stochastiques. Elles ont été activement étudiées dès la fin des années 1960. Aujourd'hui, leurs caractéristiques fondamentales (propriétés décorrélatives...) ainsi que les problèmes de leur implémentation (effet d'échantillonnage,

erreurs d'arrondi...) sont connus. Les TU sont des transformations qui permettent la représentation d'une fonction  $u(m,n)$  (l'image dans notre cas) dans un espace vectoriel défini par des fonctions de base. Elles ont la particularité de préserver toute l'information initiale. Si les fonctions de base sont notées  $a_{k,l}(m,n)$ , une TU  $v$  de la fonction  $u$  s'exprime par :

$$v(k,l) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} u(m,n) a_{k,l}(m,n), \quad 0 \leq k \leq M-1, 0 \leq l \leq N-1 \quad (1)$$

Si ces fonctions de base  $a_{k,l}$  sont orthonormales, il peut être facilement montré que n'importe quelle somme tronquée de la forme : (dans cette expression et dans la suite, \* dénote le complexe conjugué).

$$u_{P,Q}(m,n) = \sum_{k=0}^{P-1} \sum_{l=0}^{Q-1} v(k,l) a_{k,l}^*(m,n) \quad P \leq N, Q \leq M \quad (2)$$

Sera une approximation optimale au sens des moindres carrés de  $u$ , qui minimise :

$$\sigma_e^2 = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} [u(m,n) - u_{P,Q}(m,n)]^2 \quad (3)$$

Partant de ces considérations, les transformées unitaires deviennent un outil puissant puisqu'elles permettent théoriquement de synthétiser de façon complète l'information comprise dans l'image de la forme. Parmi les TU les plus usitées (qui correspondent donc à différentes familles de fonctions  $a_{k,l}(m,n)$ ), citons la transformée de Hadamard, les transformées en sinus et cosinus discrètes ou encore la transformée de Karhunen Loeve. Les deux dernières transformées ont souvent été utilisées pour la compression d'images, mais aussi pour la reconnaissance de formes. L'invariance au changement d'échelle peut être obtenue par normalisation des formes. En revanche, quelle que soit la TU utilisée parmi celles citées ci-dessus, aucune n'admet la propriété d'invariance à la rotation. Cela implique donc d'apprendre au système de reconnaissance les valeurs issues de toutes les orientations possibles des formes. Notons aussi que pour chaque TU, il existe une transformation inverse permettant le retour au signal image. Le lecteur intéressé pourra se reporter aux références [Andrews 1971] et [Gonzalez 1992] pour plus de détails concernant l'utilisation de telles transformées.

### 2.2.2.3. Les moments géométriques invariants

Depuis les débuts de la reconnaissance de formes, les modèles basés sur la théorie des moments sont certainement les plus utilisés pour la reconnaissance invariante par rotation [Dudani 1977][Abu Mostafa 1984][Resis 1991][Li 1992] [Prokop 1992]... Nous leur accordons donc une grande importance dans le cadre de cette synthèse. L'initiateur de ces approches est M. K. Hu, dont la référence [Hu 1962] est incontournable pour quiconque s'attache au problème de la reconnaissance de forme invariante. Les contributions de cet auteur sont issues de l'association de la théorie des moments avec des travaux menés sur les polynômes algébriques par des mathématiciens du XIXème siècle (Boole, Casey et Sylvester...).

Les moments géométriques sont des valeurs numériques paramétrées par leur ordre  $(p+q)$ , issues du calcul de la double intégrale (4), dans laquelle les  $I(x,y)$  sont les niveaux de gris des pixels de l'image.

$$m_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I(x,y) x^p y^q dx dy \quad p \in \mathbb{N}^+, q \in \mathbb{N}^+ \quad (4)$$

Dans son article, M. K. Hu démontre l'unicité de ces moments, en postulant que l'ensemble des  $m_{pq}$  est uniquement déterminé par les  $I(x,y)$  et qu'inversement une image  $I(x,y)$  est uniquement déterminée par ses moments  $m_{pq}$ . Néanmoins, la précision de la description des formes, comme celle de la reconstruction qu'elle permet, est liée à l'ordre  $(p+q)$  utilisé. Plus l'ordre utilisé est grand, plus la description de la forme est fine mais plus les calculs à mettre en œuvre sont lourds et entachés d'erreurs d'approximation. Une reconstruction complète exige quant à elle un ordre infini. Parmi les moments d'ordre faible les premiers ont un sens intuitif. Ainsi, le moment d'ordre 0,  $m_{00}$ , représente le poids total de l'image. Les moments d'ordre 1,  $m_{10}$  et  $m_{01}$ , permettent quant à eux de déterminer les coordonnées du centre de gravité  $(x_0, y_0)$  de l'image, en utilisant (5). Enfin, les moments d'ordre 2, c'est à dire  $m_{11}$ ,  $m_{02}$  et  $m_{20}$ , caractérisent la taille et l'orientation de l'image.

$$x_0 = \frac{m_{10}}{m_{00}} \quad y_0 = \frac{m_{01}}{m_{00}} \quad (5)$$

Si seuls les moments jusqu'à l'ordre 2 sont utilisés, une reconstruction de l'image originale amènera à une ellipse, illustrée en Figure 2.4. Les paramètres  $a$ ,  $b$  et  $\phi$  sur cette figure s'expriment par (6) et (7).

$$a = \left( \frac{m_{20} + m_{02} + \left[ (m_{20} - m_{02})^2 + 4m_{11}^2 \right]^{1/2}}{m_{00}/2} \right)^{1/2} \quad b = \left( \frac{m_{20} + m_{02} - \left[ (m_{20} - m_{02})^2 + 4m_{11}^2 \right]^{1/2}}{m_{00}/2} \right)^{1/2} \quad (6)$$

$$\phi = (1/2) \tan^{-1} \left( \frac{2m_{11}}{m_{20} - m_{02}} \right) \quad (7)$$

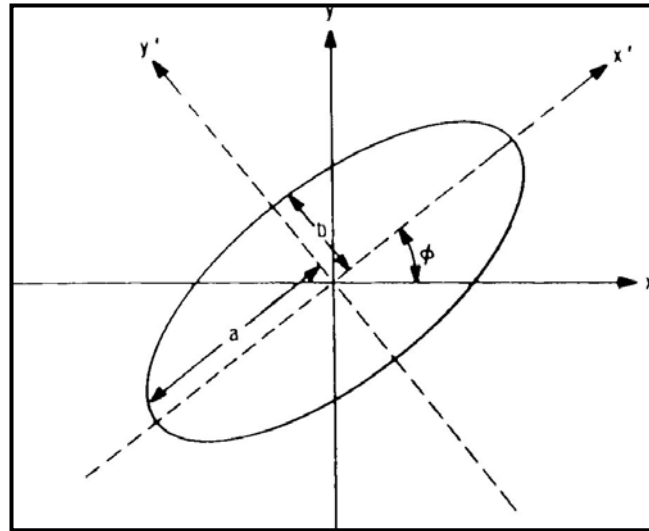


Figure 2.4 : Illustration d'une reconstruction d'ordre 2 à partir des moments de l'image

Les moments définis ci-dessus ne sont en aucune façon invariants, ni en translation, ni en rotation, ni en échelle. Pour obtenir l'invariance en translation, M. K. Hu définit les moments centraux  $M_{pq}$ , qui sont exprimés par rapport au centre de gravité de l'image :

$$M_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I(x, y)(x - x_0)^p (y - y_0)^q dx dy \quad p \in \mathbb{Z}^+, q \in \mathbb{Z}^+ \quad (8)$$

Concernant les changements d'échelle, une simple normalisation permet de s'affranchir de ce problème. Plusieurs alternatives sont possibles : une normalisation par le poids total de l'image (concrétisé par  $M_{00}$ ) ou par le rayon de giration de la forme, défini par  $(M_{02} + M_{20})^{1/2}$ . Sont issus de telles normalisations les moments centraux normalisés  $\mu_{pq}$ , qui sont invariants aux changements d'échelle (nous les présentons ici dans le cas d'une normalisation par le poids de l'image) :

$$\mu_{pq} = \frac{M_{pq}}{M_{00}^{1+\frac{p+q}{2}}} \quad \text{aussi noté } \frac{M_{pq}}{M_{00}^\gamma} \quad \text{avec } \gamma = \frac{p+q}{2} + 1 \quad (9)$$

Soit un facteur d'échelle  $\alpha$  appliqué à l'image, alors :  $x' = \alpha x$  et  $y' = \alpha y$  et il peut être montré que :

$$\mu'_{pq} = \frac{(M_{pq}')}{(M_{00}')^{\frac{(p+q)}{2}+1}} = \frac{M_{pq} \alpha^{p+q+2}}{(\alpha^2 M_{00})^{\frac{(p+q)}{2}+1}} = \frac{M_{pq}}{(M_{00})^{\frac{(p+q)}{2}+1}} = \mu_{pq} \quad (10)$$

Puisque :

$$M'_{00} = \alpha^2 \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} I(x, y) dx dy = \alpha^2 M_{00} \quad (11)$$

Les moments invariants réguliers (RMI), peuvent alors être extraits, par l'intermédiaire de :

$$(RMI)_{jk} = \sum_{r=0}^j \sum_{s=0}^k (-1)^{k-s} \binom{j}{r} \binom{k}{s} (\cos \theta)^{j-r+s} \times (\sin \theta)^{k+r-s} \times \mu_{(j+k-r-s), (r+s)} \quad (12)$$

$\binom{j}{s}$  étant le nombre de combinaisons de  $s$  éléments parmi  $j$

Dans cette expression,  $\theta$  est l'angle estimé à partir des moments du second ordre par :

$$\theta = \frac{1}{2} \tan^{-1} \left[ \frac{2M_{11}}{M_{20} - M_{02}} \right] \quad (13)$$

Cette estimation de  $\theta$  pose un certain nombre de problèmes, surtout lorsque la forme présente des symétries. C'est pourquoi de nombreux auteurs se sont aussi penchés sur l'utilisation des moments centraux normalisés pour développer d'autres modèles appelés moments invariants, exploitant la théorie des invariants algébriques. Nous examinons ci-dessous les deux études les plus connues.

#### i. Les moments invariants de Hu

A partir des moments centrés normalisés, M. K. Hu montra en 1962 [Hu 1962], qu'il est possible d'extraire des moments invariants aux rotations en utilisant les polynômes algébriques  $I$  définis par Boole, Casey et Sylvester :

$$I_{p-r,r} = \sum_{l=0}^r -i^l \binom{p-2l}{l} \sum_{k=0}^r \binom{r}{k} \mu_{p-2k-l, 2k+l} \quad \text{avec } (p-2r) > 0, i^2 = -1 \quad (14)$$

En notant  $\theta_{p,r}$  l'argument du nombre complexe  $I_{p,r}$ , les moments invariants obtenus valent alors :

$$(HMI)_0 = |I_{p0}|^2 \quad (15)$$

$$(HMI)_r = |I_{p-r,r}|^2$$

$$(HMI)_{p/2+r} = 2|I_{p-r,r}||I_{r-1,p-r+1}| \times \cos(\theta_{p-r,r} + \theta_{r-1,p-r+1})$$

Avec  $p \in \mathbb{N}^{+*}$  et  $p-2r > 0$

Dans son article précurseur [Hu 1962], M. K. Hu exploite uniquement les 7 premiers de ces moments, qui correspondent à :

- Pour le second ordre :

$$(HMI)_1 = I_{1,1} = \mu_{02} + \mu_{20} \quad (16)$$

$$(HMI)_2 = |I_{2,0}|^2 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2 \quad (17)$$

- Pour le troisième ordre

$$(HMI)_3 = |I_{3,0}|^2 = (\mu_{30} - 3\mu_{12})^2 + (3\mu_{21} - \mu_{03})^2 \quad (18)$$

$$(HMI)_4 = |I_{2,1}|^2 = (\mu_{30} + \mu_{12})^2 + (\mu_{21} + \mu_{03})^2 \quad (19)$$

$$(HMI)_5 = 2|I_{3,0}||I_{2,1}|^2 \cos(\theta_{3,0} - 3\theta_{2,1}) \quad (20)$$

$$= (\mu_{30} - 3\mu_{12})(\mu_{30} + \mu_{12}) \left[ (\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2 \right]$$

$$+ (3\mu_{21} - \mu_{03})(\mu_{21} + \mu_{03}) \left[ 3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2 \right]$$

$$(HMI)_6 = 2|I_{2,0}||I_{2,1}|^2 \cos(\theta_{2,0} - 2\theta_{2,1}) \quad (21)$$

$$= (\mu_{20} - \mu_{02}) \left[ (\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2 \right] + 4\mu_{11}(\mu_{30} + \mu_{12})(\mu_{21} + \mu_{03})$$

$$(HMI)_7 = |I_{40}|^2 = (3\mu_{21} - \mu_{03})(\mu_{30} + \mu_{12}) \left[ (\mu_{30} + \mu_{12})^2 - 3(\mu_{21} + \mu_{03})^2 \right] \quad (22)$$

$$- (\mu_{30} - 3\mu_{12})(\mu_{12} + \mu_{03}) \left[ 3(\mu_{30} + \mu_{12})^2 - (\mu_{21} + \mu_{03})^2 \right]$$

Les 6 premiers moments sont invariants aux translations, aux changements d'échelle, aux rotations ainsi qu'aux réflexions. Or, l'invariance aux réflexions peut être problématique quand il s'agit de reconnaître des images « miroirs ». C'est pourquoi M. K. Hu exploite aussi dans son système le moment  $(HMI)_7$  qui n'est pas invariant aux réflexions. Ce dernier change de signe lorsqu'une telle transformation est appliquée à l'image et permet donc de détecter celle-ci.

En terme de validation, si beaucoup d'auteurs ont présenté le résultat de tests menés sur des images binaires (comme nous le verrons en 2.2.3.4), nous n'avons pas connaissance de travaux extrayant directement les invariants de Hu à partir d'image en niveaux de gris et ce, malgré la possibilité théorique qui découle des définitions ci-dessus. Notons aussi concernant les moments de Hu que leur extraction est très coûteuse en terme de temps de calcul. Une telle complexité a poussé de nombreuses équipes à mener des travaux pour l'optimisation des calculs. Le lecteur intéressé par ces considérations

trouvera une bonne synthèse et une étude comparative de ces travaux dans [Lefrere 1993], cette étude montre la supériorité de la méthode de [Dai 1992].

**ii. Les moments de Bamieh et de Figueiredo**

Au même titre que les moments de Hu, les moments de Bamieh, définis dans [Bamieh 1986], font partie de la famille des moments algébriques invariants. Ils sont essentiellement utilisés pour la modélisation de formes 3D. La principale caractéristique de ce modèle numérique est sa taille beaucoup plus réduite que celui utilisé dans [Hu 1962] puisqu'il consiste en l'utilisation d'uniquement 4 valeurs invariantes. Ces valeurs sont issues de la théorie des tenseurs (23), dont la combinaison conduit à l'obtention des moments invariants de Bamieh (*BMI*) (24)-(27). Nous exprimons ici ces *BMI* fonction des  $\mu$  définis dans (9).

$$M^{ijk\dots} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^i x^j x^k \dots f(x^1, x^2) dx^1 dx^2, \quad i, j, k \in \{1, 2\} \quad (23)$$

$$(BMI)_1 = \mu_{02}\mu_{20} - \mu_{11}^2 \quad (24)$$

$$(BMI)_2 = (\mu_{03}\mu_{30} - \mu_{21}\mu_{12})^2 - 4(\mu_{03}\mu_{12} - \mu_{21}^2)(\mu_{21}\mu_{30} - \mu_{12}^2) \quad (25)$$

$$(BMI)_3 = \mu_{40}\mu_{04} - 4\mu_{31}\mu_{13} + 3\mu_{22}^2 \quad (26)$$

$$(BMI)_4 = \mu_{40}\mu_{22}\mu_{04} - 2\mu_{31}\mu_{22}\mu_{13} - \mu_{40}\mu_{13}^2 - \mu_{04}\mu_{31}^2 - \mu_{22}^2 \quad (27)$$

**2.2.2.4. Les moments invariants de Zernike et leurs dérivés**

Le modèle le plus fréquemment utilisé dans la littérature pour la reconnaissance de formes invariante aux rotations et aux changements d'échelle est sans conteste celui basé sur les moments de Zernike. De très nombreux auteurs se sont penchés sur cette approche en s'inspirant des travaux initiaux de [Teague 1980]. Nous présentons ci-après en détail la contribution de ce dernier, en nous attachant à la définition des moments de Zernike. Puis nous décrivons plus succinctement les différentes contributions ayant pour but de calculer des invariants issus de ces moments ainsi que les problèmes de mise en œuvre.

**i. Les polynômes et moments de Zernike**

La base théorique de ces approches consiste à utiliser un ensemble de polynômes orthogonaux (les polynômes de Zernike [Zernike 1938]), qui constituent un espace vectoriel dans lequel l'image de la forme est projetée. Cela permet de s'affranchir du problème de normalisation rencontré dans le cas des moments de Hu, mais aussi d'aboutir à une description plus précise des formes, par utilisation d'ordres plus élevés, en maintenant les propriétés d'invariance. Les polynômes de Zernike sont définis de la façon suivante :

$$\begin{aligned} V_{nl}(x, y) &= V_{nl}(\rho \cos \theta, \rho \sin \theta) \\ &= R_{nl}(\rho) \exp(il\theta) \end{aligned} \quad (28)$$

avec :



$$i^2 = -1 \quad n \in \mathbb{Z}, l \in \mathbb{Z} \quad \text{avec } |l| \leq n \quad \text{et } (n - |l|) \text{ pair}$$

et :

$$R_{nl}(\rho) = \sum_{s=0}^{(n-|l|)/2} \frac{(-1)^s \rho^{n-2s} (n-s)!}{s! \left(\frac{n+|l|}{2} - s\right)! \left(\frac{n-|l|}{2} - s\right)!} \quad (29)$$

Ces polynômes  $V_{nl}(x,y)$  sont orthogonaux à l'intérieur du cercle unité et vérifient la relation :

$$\iint_{x^2+y^2 \leq 1} [V_{nl}(x,y)]^* [V_{mk}(x,y)] dx dy = \frac{\pi}{(n+1)} \delta_{mn} \delta_{kl} \quad \text{où } \delta_{ij} = \begin{cases} 1 & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases} \quad (30)$$

En projetant une image  $f(x,y)$  sur l'espace défini par les polynômes  $V_{nl}(x,y)$ , la définition des moments complexes de Zernike est alors dérivée, en appliquant :

$$A_{nl} = \frac{(n+1)}{\pi} \iint_{x^2+y^2=1} f(x,y) [V_{nl}(x,y)]^* dx dy = (A_{n,-l})^* \quad (31)$$

A la suite d'une telle transformation, l'image originale peut être reconstruite, par calcul de la transformée inverse, en utilisant :

$$f(x,y) = \lim_{N \rightarrow \infty} \sum_{n=0}^N \sum_l A_{nl} V_{nl}(x,y) \quad \text{avec } n - |l| \geq 0 \quad \text{et } \text{pair} \quad (32)$$

Si les intervalles de variation de  $n$  et  $l$  sont infinis, l'image reconstruite sera rigoureusement la même que l'image ayant permis l'extraction des moments. Dans le cas contraire, il s'agira d'une approximation de l'image : un modèle approché.

Notons qu'il est également possible d'exprimer les différents moments de Zernike à partir des moments centrés normalisés, présentés en (9). Pour ce faire, il est nécessaire d'écrire les polynômes sous leur forme radiale  $R_{nl}(\rho)$ , en les décomposant en puissances de  $\rho$  :

$$R_{nl}(\rho) = \sum_{k=l}^n B_{nlk} \rho^k \quad \text{avec } (n-k) \text{ pair} \quad (33)$$

avec :

$$B_{nlk} = \frac{(-1)^{(n-k)/2} [(n+k)/2]!}{[(n-k)/2]! [(l+k)/2]! [(k-l)/2]!} \quad (34)$$

avec  $(l-k)$  et  $(n-k)$  pairs

En utilisant ces  $B_{nlk}$ , l'expression des moments de Zernike devient alors :

$$A_{nl} = [(n+1)/\pi] \sum_{k=l}^n \sum_{j=0}^q \sum_{m=0}^l (-i)^m \binom{q}{j} \binom{l}{m} B_{nlk} * \mu_{k-2j-l+m, 2j+l-m} \quad \text{avec } q = (k-l)/2 \quad (35)$$

A titre d'illustration, nous proposons ci-après l'expression de quelques moments issus de l'application de (35) :

$$\begin{aligned}
 A_{00} &= \mu_{00} / \pi = 1 / \pi \\
 A_{11} &= A_{1,-1} = 0 \\
 A_{22} &= 3(\mu_{02} - \mu_{20} - 2i\mu_{11}) / \pi \\
 A_{20} &= 3(2\mu_{20} + 2\mu_{02} - 1) / \pi \\
 A_{33} &= 4[\mu_{03} - 3\mu_{21} + i(\mu_{30} - 3\mu_{12})] / \pi \\
 A_{31} &= 12[\mu_{03} + \mu_{21} - i(\mu_{30} + \mu_{12})] / \pi \\
 A_{44} &= 5[\mu_{40} - 6\mu_{22} + \mu_{04} + 4i(\mu_{31} - \mu_{13})] / \pi \\
 A_{42} &= 5\{4(\mu_{04} - \mu_{40}) + 3(\mu_{20} - \mu_{02}) - 2i[4(\mu_{31} + \mu_{13}) - 3\mu_{11}]\} / \pi \\
 A_{40} &= 5[6(\mu_{40} + 2\mu_{22} + \mu_{04}) - 6(\mu_{20} + \mu_{02}) + 1] / \pi
 \end{aligned} \tag{36}$$

Les modules des  $A_{nl}$  ainsi définis sont invariants à toute rotation appliquée à  $f$  mais l'information de phase est alors perdue. Les invariances à la translation ainsi qu'au changement d'échelle peuvent être obtenues respectivement par décalage et normalisation. Cette dernière peut être effectuée en utilisant les moments réguliers du premier ordre, comme pour les moments de Hu. Dans la partie suivante, nous décrivons différentes méthodes consistant à extraire des invariants basés sur ces moments, en gardant l'information de phase.

## ii. Extraction d'invariants à partir des moments de Zernike

De nombreux auteurs ont dérivé des invariants basés sur les moments de Zernike. Présenter de façon exhaustive ces études n'étant pas possible, nous listons uniquement certaines de ces contributions. Nous détaillons d'abord celle de Teague qui est à la base des autres travaux, puis nous listons les autres contributions dans ce domaine, en essayant d'expliquer les différences par rapport à l'approche de Teague. Signalons aussi les nombreuses études telles que [Prokop 1992], [Mukundan 1995], [Liao 1996, 1998], [Bailey 1996] ou [Mamistvalov 1998], qui ont eu pour but d'optimiser l'extraction du modèle de Zernike, que ce soit en terme de complexité ou en terme de performances ou pour apporter une validation théorique des approches. Nous ne détaillerons pas ces contributions.

### • Les invariants de Zernike selon Teague [Teague 1980]

Le premier concerné par l'extraction d'invariants à partir des moments de Zernike est bien sûr Teague lui-même [Teague 1980]. Partant de l'équation (31), l'auteur montre que si une forme subit une rotation d'angle  $\theta$ , alors les moments  $A'_{nl}$  issus de cette nouvelle forme peuvent être exprimés en fonction des  $A_{nl}$  issus de l'image originale par :

$$A'_{nl} = A_{nl} \exp(-il\theta) \tag{37}$$

A partir de cette constatation, l'auteur dérive la formulation générale de ce que nous appelons les invariants de Zernike selon Teague ((38)-(40)), qui seront notés  $TZMI$  par la suite.

$$(TZMI)_{n0} = A_{n0} \tag{38}$$

$$(TZMI)_{nl} = |A_{nl}|^2 \tag{39}$$

$$(TZMI)_{nz} = [A_{nl}^* (A_{mh})^p] - [A_{nl}^* (A_{mh})^p]^* \tag{40}$$

Avec  $h \leq l, p \geq 1, p = l/h, (l \bmod h) = 0$ , et  $z = p + l + h$

Les équations (38) et (39) expriment les invariants primaires alors que (40) exprime les invariants secondaires. Ces invariants secondaires sont invariants aux rotations. Notons qu'ils peuvent être rendus invariants aux réflexions si un « + » est utilisé entre les deux produits complexes à la place du « - ». Parmi l'infinité d'invariants pouvant être calculés à partir de (39), il est nécessaire de choisir, pour modéliser une forme, ceux qui sont fonctionnellement indépendants. Pour ce faire, Teague utilise une méthode basée sur des essais/erreurs et l'aléatoire. Les invariants allant jusqu'au 7<sup>ème</sup> ordre issus de cette approche sont donnés en Annexe de [Belkasim 1991].

- **Les invariants de Zernike de Belkasim [Belkasim 1991]**

Dans l'article [Belkasim 1991], un nouveau jeu d'invariants de Zernike est également proposé. Ces invariants, qui seront notés *BZMI* par la suite, s'expriment par :

$$(BZMI)_{n0} = A_{n0} \quad (41)$$

$$(BZMI)_{nl} = |A_{nl}| \quad (42)$$

$$(BZMI)_{n,n+z} = [A_{mh}^* (A_{nl})^p] - [A_{mh}^* (nl)^p]^* \quad (43)$$

Avec  $h \leq l, m \leq n, p = h/l, 0 \leq p \leq 1$ , et  $z = l/h$

Les différences avec les invariants définis par Teague peuvent facilement être remarquées. D'une part, dans (42), le module est utilisé (et non son carré), ce qui aura pour influence de diminuer l'amplitude de l'invariant si elle est supérieure à 1. Ensuite, dans (43), les contraintes sont plus strictes, ce qui aura tendance à diminuer la dynamique des invariants à cause de l'exposant  $p$  compris entre 0 et 1. Outre cette nouvelle formulation, Belkasim introduit également un nouveau procédé de normalisation des moments permettant de réduire la dynamique des valeurs avant l'extraction des invariants. Cette normalisation est basée sur les équations suivantes :

$$\tilde{A}_{nl} = A_{nl} / A_{n-2,l} \quad \text{si } A_{n-2,l} \neq 0 \quad \text{et } l < n \quad (44)$$

$$\tilde{A}_{nl} = A_{nl} \quad \text{si } A_{n-2,l} = 0 \quad \text{ou } l = n \quad (45)$$

- **Les invariants de pseudo Zernike**

Contrairement aux invariants présentés précédemment, les invariants de pseudo Zernike ne sont pas issus des polynômes de Zernike. Néanmoins, nous les présentons ici parce que leur méthode de construction, ainsi que leurs propriétés sont exactement les mêmes que dans le cas des invariants de Zernike. La différence entre les deux approches vient de l'utilisation des polynômes de pseudo Zernike qui s'expriment non pas uniquement en fonction de  $x$  et  $y$ , mais en fonction de  $x, y$ , et  $r$  comme suit :

$$W_{nl}(x, y, r) = W_{nl}(r \cos \phi, r \sin \phi, r) = S_{nl}(r) \exp(il\phi) \quad (46)$$

A partir de ces polynômes, les moments de pseudo Zernike sont calculés en utilisant :

$$T_{nl} = \frac{n+1}{\pi} \int_0^{2\pi} \int_0^{\infty} f(r \cos \phi, r \sin \phi) S_{nl}(r) \times \exp(-il\phi) r dr d\phi \quad n \in \square, \quad l \leq n \quad (47)$$

Puis, exactement comme pour les moments de Zernike, il s'agit d'extraire les invariants de pseudo Zernike (notés *PZMI*). Pour ce faire, Belkasim propose la même approche que pour les invariants de Zernike, c'est à dire l'utilisation de :

$$(PZMI)_{n0} = T_{n0} \quad (48)$$

$$(PZMI)_{nl} = |T_{nl}| \quad (49)$$

$$(PZMI)_{n,n+z} = [T_{mh}^* (T_{nl})^p] \pm [T_{mh}^* (T_{nl})^p]^* \quad (50)$$

Avec  $h \leq l$ ,  $m \leq n$ ,  $z = l/h$ ,  $p = h/l$  et  $p \in [0,1]$

### 2.2.2.5. Les invariants issus d'ondelettes

Les études citées dans cet état de l'art sont pour la plupart des travaux assez anciens qui constituent toujours des références dans le domaine. Depuis 1998, de nouvelles contributions s'intéressent à l'exploitation de la théorie des ondelettes pour la reconnaissance de formes. Une transformée en ondelettes diffère d'une transformation de Fourier par le fait qu'elle est capable de fournir une représentation à la fois fréquentielle et temporelle d'un signal 1D. Elle permet d'accomplir une analyse à la fois locale et globale d'une forme [Daubechies 1990,1992]. Elle est donc particulièrement adaptée à l'extraction d'un modèle de formes discriminant grâce à cet aspect multi-résolutions. Parmi les travaux exploitant cette théorie des ondelettes, nous présentons deux exemples d'approches.

- **L'approche de Shen [Shen 1999]**

Dans [Shen 1999], les auteurs présentent d'abord un cadre général de la reconnaissance de formes invariante à la rotation. En analysant les différents travaux présentés dans la littérature, les auteurs déduisent que toutes les approches à base de moments peuvent s'exprimer par l'intermédiaire de :

$$F_{pq} = \int S_q(r) \cdot g_p(r) r dr \quad \text{avec } S_q(r) = \int f(r \cos \theta, r \sin \theta) e^{iq\theta} d\theta \quad (51)$$

Dans cette expression,  $F_{pq}$  désigne un moment invariant d'ordre  $pq$  ( $p$  et  $q$  sont entiers), et  $g_p(r)$  désigne une fonction de la variable radiale  $r$ . A titre d'exemple, si  $g_p(r) = r^p$ , il est possible, en imposant des contraintes à  $p$  et  $q$ , d'exprimer les moments de Hu. L'auteur propose également une formulation de  $g_p(r)$  permettant le calcul des moments de Zernike, sans toutefois exprimer l'ensemble des contraintes à imposer à  $p$  et  $q$ . A partir de cette proposition de cadre unifié d'extraction de moments, les auteurs substituent à la fonction  $g_p$  une fonction de base d'ondelettes, en considérant la famille :

$$\psi_{a,b}(r) = \frac{1}{\sqrt{a}} \psi\left(\frac{r-b}{a}\right) \quad (52)$$

L'ondelette mère utilisée dans l'article est l'ondelette B-Spline Cubique, dont la formulation, proposée dans [Unser 1996], est donnée par :

$$\psi(r) = \frac{4a^{n+1}}{\sqrt{2\pi(n+1)}} \sigma_w \cos(2\pi f_0(2r-1)) \exp\left(-\frac{(2r-1)^2}{2\sigma_w^2(n+1)}\right) = g_p(r) \quad (53)$$

$n = 3 \quad a = 0,697066 \quad f_0 = 0,409177 \quad \sigma_w^2 = 0,561145$

Pour modéliser leurs formes, les auteurs utilisent alors le module de (51), qui permet d'obtenir une description des formes invariante en rotation. L'invariance au changement d'échelle est quant à elle obtenue par l'intermédiaire d'un procédé classique de normalisation par rapport aux moments réguliers.

- **L'approche de Chen [Chen 1999]**

Celle-ci diffère quelque peu de la précédente. En effet, partant des qualités et défauts de la transformée de Fourier et de la transformée en ondelettes, l'auteur combine l'utilisation des deux. Le principe utilisé peut être décrit par le synoptique sur la Figure 2.5. La première étape de cette extraction du modèle consiste à normaliser la forme et à passer celle-ci dans le domaine polaire. Pour ce faire, les auteurs utilisent un découpage en zones de leur forme, en se basant sur des cercles concentriques et sur des sondes radiales. Un moyennage dans chacune des zones permet de passer des coordonnées cartésiennes aux coordonnées polaires. Ensuite, une transformée de Fourier 1D est appliquée suivant l'axe des  $\theta$ , afin d'obtenir l'invariance à la rotation, compte tenu du théorème de transfert bien connu en analyse de Fourier. Une transformée en ondelettes est alors appliquée, cette fois suivant la variable radiale  $r$  afin de profiter de l'aspect multi-résolutions des ondelettes.

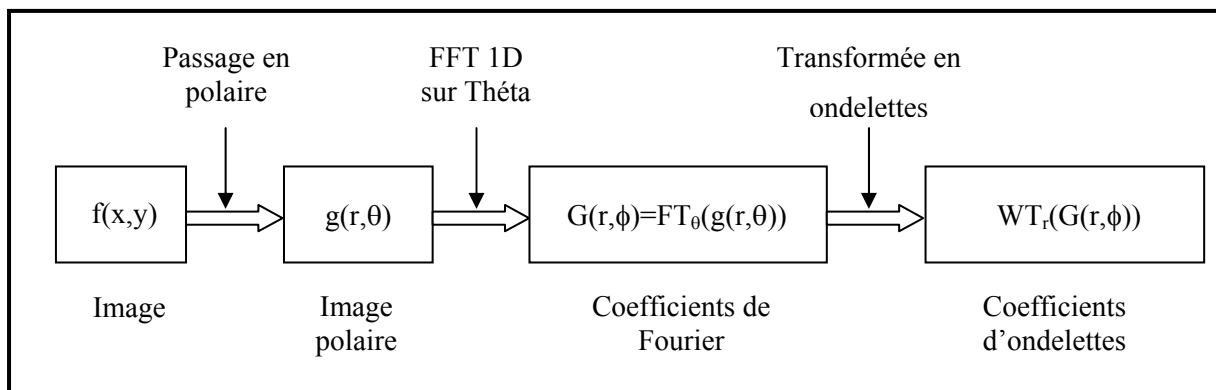


Figure 2.5 : Synoptique de l'extraction du modèle de Chen [Chen 1999]

Notons enfin concernant ces approches à base d'ondelettes une récente contribution [Lai 2001] exploitant là encore une combinaison entre transformée de Fourier et transformée en ondelettes pour la reconnaissance de visages.

### 2.2.2.6. Synthèse des modèles extraits d'images en niveaux de gris

Nous avons décrit dans cette partie de nombreux modèles dédiés à la reconnaissance de formes se basant sur l'image en niveaux de gris. Chacun de ces modèles est en principe capable de synthétiser l'information comprise dans l'image de la forme lorsque celle-ci prend ses valeurs dans le domaine [0,255]. Néanmoins, si la littérature montre souvent la possibilité théorique d'utilisation de telles méthodes, les applications basées sur de telles approches sont beaucoup plus rares. La raison majeure de cet état de fait vient du problème de la localisation des formes. En effet, si isoler une forme

représentée en noir et blanc est aisée, par l'utilisation d'un extracteur de composantes connexes, il n'en est pas de même pour les formes en niveaux de gris. Une binarisation ou une extraction des contours pourraient être envisagées pour détecter la forme. Elles permettraient alors de détecter les limites de la forme pour ensuite revenir sur l'image en niveaux de gris de celle-ci afin d'appliquer les modèles précédents. Mais une grande partie de l'intérêt des nuances de gris serait alors perdue, essentiellement au niveau de formes connectées entre elles. Une autre alternative pourrait consister à tester toutes les positions possibles de formes dans une image, mais le coût en terme de temps de calcul interdit souvent une telle approche sur de grandes images. C'est pourquoi il est difficile d'avoir un point de vue critique sur les différentes méthodes exposées ci-avant. Dans le paragraphe qui suit, nous proposerons en revanche une évaluation de ces différents modèles dans le cadre de leur utilisation sur des images binaires. Une comparaison que nous avons menée sur différentes bases de données sera également présentée en 2.4.

### **2.2.3. Les modèles issus de l'image binaire**

Dans le cadre de l'interprétation de documents techniques, il est très fréquent que, pour des raisons de stockage, les images soient acquises directement sous forme binaire. Dans le cas contraire, elles subissent souvent une binarisation dès le premier maillon de la chaîne d'interprétation, pour les raisons que nous avons explicitées ci-avant. Quel que soit le cas, la fonction  $I(x,y)$  caractérisant l'image ne prend alors que deux valeurs distinctes et le modèle utilisé pour la reconnaissance doit être extrait directement de cette image binaire. Dans cette partie, nous abordons ces différents modèles pouvant être extraits de ce type d'image. Contrairement à ce qui a été fait dans la section précédente, les résultats obtenus par les différentes contributions sont ici proposés.

#### **2.2.3.1. L'image et ses représentations dérivées**

Au même titre que pour une image en niveaux de gris, il est possible, du point de vue conceptuel, d'utiliser directement l'image binaire d'une forme comme modèle, utilisant une distance entre images pour prendre la décision. Les distances disponibles sont très nombreuses, un très bon état de l'art les concernant est disponible dans [Klette 1987]. Parmi les utilisateurs d'une telle approche, citons par exemple [Gader 1991] qui l'exploite pour la reconnaissance de caractères manuscrits. Néanmoins, si pour s'affranchir du problème de taille et de position variable, un mécanisme de normalisation est envisageable, une orientation variable des formes impliquera nécessairement d'apprendre toutes les orientations possibles des formes. Les différents verrous évoqués en 2.2.2.1 sont alors de nouveau valables. C'est pourquoi il n'y a, à notre connaissance, aucun système utilisant ce genre d'approches pour la reconnaissance de formes multi-orientées et multi-échelles.

Une autre solution basée sur l'image, comme dans le cas des images en niveaux de gris, consiste à découper celle-ci en zones, et à évaluer des densités de pixels noirs dans chacune des zones. C'est un type d'approche souvent utilisé pour la reconnaissance de la structure d'un document [Heroux 2001].

Il est parfois également exploité dans les systèmes d'OCR, comme par exemple dans le système *Calera* décrit dans [Bokser 1992]. Mais, de nouveau, ce type d'approche ne semble généralisable dans le cas de la multi-orientation et du multi-échelle que par un apprentissage de toutes les configurations. Notons tout de même qu'il est possible d'envisager l'utilisation de zones basées sur des anneaux plutôt que sur des rectangles pour gérer le problème de multi-orientation. Dans un tel cas, un tri des densités de pixels permet une extraction de modèle invariant à la rotation. Une illustration du type de figure à utiliser est donnée en Figure 2.7.

### 2.2.3.2. Les transformées unitaires

Si aucune équipe n'a, à notre connaissance, exploité la possibilité théorique d'appliquer les transformées unitaires à des images en niveaux de gris pour modéliser les formes, il n'en est pas de même dans le cas des images binaires. En effet, la théorie présentée en 2.2.2.2 peut parfaitement s'appliquer à des images dont les pixels ne prennent que deux valeurs. C'est par exemple le cas du système de reconnaissance d'écriture manuscrite NIST, présenté dans [Garris 1994], qui utilise la transformée de Karhunen-Loeve pour extraire un modèle de formes binaires. Les performances de ce système du domaine public sont réputées comme étant très bonnes, ce qui prouve le potentiel intérêt de telles approches. Toutefois, là encore le problème de la multi-orientation ne peut être pris en compte que par un apprentissage de toutes les configurations d'orientation.

### 2.2.3.3. Les histogrammes de projection

Les histogrammes de projection sont depuis très longtemps utilisés dans le domaine de la reconnaissance de formes [Glauberman 1956]. Leur principe est de sommer le nombre de pixels noirs de chacune des lignes (respectivement des colonnes) de l'image binaire de la forme, comme le montre la partie gauche de la Figure 2.6. Les amplitudes de chacune des barres des histogrammes peuvent alors être utilisées comme modèle de la forme.

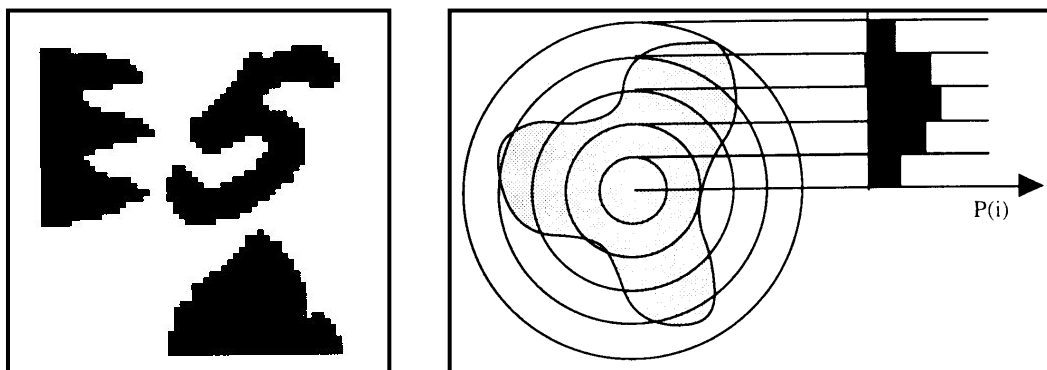


Figure 2.6 : Illustration d'histogrammes de projection : à gauche, [Trier 96], à droite [Lee 1991]

Comme dans le cas du découpage en zones, un tel modèle peut facilement être rendu invariant au changement d'échelle par normalisation, mais les changements d'orientation constituent un problème. En effet, les histogrammes de projection sont particulièrement sensibles à une légère modification de

la rotation des formes. C'est pourquoi s'ils étaient auparavant utilisés pour modéliser les formes, une telle approche est aujourd'hui plutôt utilisée pour segmenter les caractères horizontalement, pour détecter les lignes de texte ou encore pour redresser un document [Kasturi 1990].

Notons tout de même la contribution de Lee [Lee 1991] qui utilise le principe des histogrammes de projections, mais en basant celui-ci sur des anneaux plutôt que sur des zones rectangulaires, afin d'obtenir l'invariance à la rotation. La méthode est d'un point de vue théorique opérationnelle mais ne permet pas l'obtention d'une bonne discrimination et semble très sensible aux bruits.

#### **2.2.3.4. Les moments géométriques invariants**

Dans cette partie, nous abordons les modèles basés sur la théorie des moments algébriques présentés en 2.2.2.3, lorsque ceux-ci sont appliqués à des images binaires. Le principe de l'extraction de ces modèles est exactement le même que celui évoqué en 2.2.2.3 ; c'est pourquoi nous ne le reprenons pas ici et nous nous focalisons plutôt sur l'aspect évaluation. Dans le contexte de la reconnaissance de formes binaires, de nombreuses équipes se sont attachées à tester l'approche de M. K. Hu, à commencer par l'auteur lui-même. En guise de validation, ce dernier montre dans [Hu 1962] la preuve de l'invariance de ses moments aux rotations, translations et changements d'échelle sur des formes géométriques binaires simples (carrés pleins ou creux, losanges...). Il présente aussi dans cet article quelques résultats sur des caractères alphanumériques. Si la preuve de l'invariance est montrée, les résultats quantitatifs présentés sont en revanche peu éloquentes. Seuls les deux premiers moments sont utilisés pour traiter des bases très réduites, et il est donc difficile d'en tirer des conclusions quant à l'efficacité du modèle. La validité de l'approche a en revanche été montrée par d'autres auteurs. Parmi ceux-ci, citons les travaux de [Khotanzad 1988] qui obtient des taux de reconnaissance allant jusqu'à 98% sur des caractères majuscules par utilisation des moments de Hu. Les mêmes moments sont également utilisés dans [El-Khaly 1990] pour la reconnaissance de caractères arabes, ils permettent l'obtention d'un score de 90% de reconnaissance. Dans ces deux études, les bases d'apprentissage et de test sont toutefois extrêmement réduites et de ce fait, les résultats annoncés sont peu fiables. Notons aussi que les moments de Hu sont comparés à d'autres approches dans [Belkasim 1991]. Nous présentons cette étude comparative ci-après, dans le cadre des travaux basés sur les moments de Zernike. Enfin, citons la récente contribution de l'équipe de Balslev [Balslev 2000] concernant le calcul des moments centrés, préalable à celui des moments de Hu. Les auteurs introduisent une fonction de pondération dont le but est de diminuer l'influence des pixels éloignés du centre de gravité. Celle-ci permet de rendre la méthode moins sensible au bruit, puisque celui-ci apparaît en général aux frontières des formes. Toutefois, aucun résultat quantitatif n'est présenté dans cette contribution, même si les auteurs obtiennent de bons résultats pour la reconnaissance de briques de LEGO® dans le cadre d'une application de robotique. Notons aussi que les auteurs abordent le



problème de l'estimation de l'orientation, mais que celle-ci est soumise aux résultats de la reconnaissance.

### 2.2.3.5. Les moments de Zernike

Comme dans le cas des moments de Hu, le principe d'application des moments de Zernike ayant été largement décrit en 2.2.2.4, nous n'y reviendrons pas ici. Il est rigoureusement le même que dans le cas d'images en niveaux de gris et nous nous contenterons donc dans ce paragraphe de faire une synthèse des travaux ayant exploité une telle approche sur des images binaires. Là encore, les premiers résultats émanent évidemment du concepteur de l'approche, à savoir R. Teague [Teague 1980]. Pour valider son modèle, celui-ci a testé son approche sur quelques lettres majuscules présentées dans un nombre fini d'orientations. Ces tests utilisent un modèle constitué de 43 valeurs numériques correspondant aux premiers moments invariants définis en 2.2.2.4. La stabilité de ces valeurs est montrée vis-à-vis de la rotation puisque un écart maximal de 20% autour de la moyenne est mesuré.

Parmi les travaux ultérieurs, Khotanzad teste dans [Khotanzad 1990a] et [Khotanzad 1990b] l'approche développée par Teague sur ses propres données. Le passage des moments de Hu aux moments de Zernike semble montrer la supériorité des seconds. Les auteurs annoncent des scores de reconnaissance sur une base conséquente allant jusqu'à 94% alors que les moments de Hu n'obtiennent que 56% sur cette même base. Les auteurs montrent également que la reconstruction des images à partir des moments de Zernike jusqu'à l'ordre 12 engendre une erreur sur l'image résultat inférieure à 10% du poids total de l'image.

Une autre étude très intéressante, qui constitue une référence incontournable de la littérature, est l'étude comparative menée dans [Belkasim 1991]. L'auteur y compare les moments invariants réguliers (*RMI*, (12)), les moments de Hu (*HMI*, (16)-(22)), les moments de Bamieh (*BMI* (24)-(27)), les moments de Zernike définis par Teague (*TZMI*, (38)-(40)), les moments invariants de Zernike définis par Belkasim (*BZMI*, (41)-(43)) et les moments de pseudo Zernike (*PZMI* (48)-(50)). Enfin, l'auteur ajoute à ces invariants classiques des versions normalisées des moments de Zernike (notés *NBZMI*), et de pseudo Zernike (notés *NPBZMI*). Ceux-ci sont basés sur un calcul récursif. La méthode d'obtention de ces invariants est précisément décrite dans [Belkasim 1991]. Les caractéristiques des données utilisées pour ce comparatif sont résumées dans le Tableau 2.1. Dans les deux cas, les images sont acquises en niveaux de gris, mais elles sont binarisées avant l'extraction du modèle et rentrent donc parfaitement dans le cadre de cette partie de la synthèse. La méthode adoptée pour la binarisation est décrite dans [Belkasim 1988].

	Nombre de classes	Nombre élément / classe	Particularités
Images d'avions	18	16	Multi-orientation
Chiffres manuscrits	10	32	Ecriture libre non multi-orientée

*Tableau 2.1 : Les données utilisées dans le comparatif de Belkasim [Belkasim 1991]*

Les différents moments sont calculés en exploitant les moments centraux normalisés  $\mu_{pq}$  jusqu'à l'ordre 7. L'analyse des résultats montre que de façon générale, les *BZMI* et plus particulièrement leur version normalisée *NBZMI* obtiennent les meilleurs résultats, tant du point de vue de la discrimination que du point de vue du taux d'erreur. Cette constatation reste valable quel que soit l'ordre pris en compte et quel que soit le jeu de données. Notons aussi que ces tests ont également été réalisés en présence d'un bruit blanc Gaussien additionnel et que les résultats sont similaires, surtout en ce qui concerne la supériorité des *BZMI* et des *NBZMI*. Cette étude très complète a constitué pour nous une base de travail. Elle nous a permis d'avoir une référence quant aux méthodes de la littérature. C'est ce qui nous a motivé pour l'utilisation des *NZMI* dans le cadre de nos travaux.

Notons aussi que si la plupart des références citées ci-dessus sont anciennes, de nouvelles contributions concernant les moments de Zernike paraissent régulièrement dans la littérature. Citons par exemple les récents travaux de Zhenjiang [Zhenjiang 2000], dont l'article présente une nouvelle application de reconnaissance de formes. Le but de celle-ci est d'analyser des images binarisées de roses pour en extraire des paramètres permettant de mettre des copyrights sur les variétés créées. En utilisant un unique moment, les auteurs annoncent des résultats satisfaisants, supérieurs à des résultats obtenus par les descripteurs de Fourier que nous présenterons en 2.2.4.4.

### 2.2.3.6. Les représentations à base d'ondelettes

Comme pour les moments géométriques et pour les invariants de Zernike, ce paragraphe est dédié à la présentation des résultats obtenus lorsque la théorie présentée en 2.2.2.5 est appliquée à des images binaires. En effet, dans les deux contributions évoquées dans ce paragraphe, les tests effectués par les auteurs le sont sur des images binaires.

#### *i. L'approche de Shen [Shen 1999]*

Dans [Shen 1999], les résultats qui sont annoncés par les auteurs le sont sur 4 bases différentes :

- Une base constituée des formes « 1 » et « | »
- Une base constituée de 2 formes de synthèse
- Une base constituée des 26 lettres majuscules de l'alphabet,
- Une base constituée d'images de pinces

Dans chacun des cas, les auteurs déclinent les formes suivant 60 orientations et échelles différentes, en générant aléatoirement les orientations, les échelles et un bruit perturbateur. Pour chacune de ces bases, les auteurs associent leur modèle à base d'ondelettes à un processus de sélection des primitives pertinentes sur la base d'apprentissage. Cela permet d'adapter le modèle au type de données traitées,

en ajustant le nombre de primitives utilisé. Dans chacun des cas, les auteurs obtiennent un taux de reconnaissance de 100%. Néanmoins, si chacune des bases permet d'observer le comportement du modèle face à un problème particulier (séparation de deux classes proches, comportement vis à vis du bruit), cette étude ne présente pas de tests représentatifs d'un contexte multi-classes avec une forte variabilité de formes. Les résultats comparatifs présentés illustrent en effet la relative facilité des problèmes abordés puisque les moments de Zernike, s'ils ne sont pas tout à fait aussi performants obtiennent également de très bons taux de reconnaissance.

*ii. L'approche de Chen [Chen 1999]*

Différentes ondelettes (Haar, Daubechies, Coiflet, symmlet) ont été testées dans cette contribution. Quelle que soit la famille utilisée, l'approche permet l'obtention d'un taux de reconnaissance de 100%. Néanmoins, là encore, la base de données a été constituée synthétiquement, par rotation et dilatation de formes de bases, en l'occurrence des caractères chinois. La méthode de rotation utilisée pour la génération de la base n'est pas explicitée. Celle-ci revêt pourtant beaucoup d'importance. En effet, nous avons constaté dans le cadre de premiers travaux sur les ondelettes que si des données sont générées synthétiquement en utilisant l'espace polaire, les résultats sont fortement biaisés puisque la même approche est utilisée en synthèse et en reconnaissance. De plus, dans [Chen 1999], aucun résultat sur des données réelles n'est présenté.

**2.2.3.7. Les représentations à base de primitives circulaires**

Le cercle étant par définition une figure géométrique invariante à la rotation, la littérature utilise souvent les primitives circulaires quand l'objectif est d'avoir un modèle invariant à la rotation [Bebis 1992]. La plupart de ces méthodes utilisent le barycentre comme référence, afin de rendre le problème invariant à la translation. L'invariance au changement d'échelle est quant à elle obtenue en normalisant les primitives par rapport au diamètre ou au rayon de la forme à identifier.

Parmi ces approches, les descripteurs proposés dans [Taza 1989] se basent sur un ensemble de  $N$  cercles concentriques, centrés sur le barycentre de la forme, et de rayons proportionnels au rayon du cercle « maximal » (passant par le point de la forme le plus éloigné du centre de gravité). En utilisant des sondes concentriques, chaque cercle est alors divisé en  $M$  secteurs identiques. Les points d'intersections des  $M$  rayons avec les  $N$  cercles constituent alors les points échantillons de la forme. En parcourant le système de cercles dans un ordre et un sens donnés, les auteurs construisent une matrice binaire basée sur la densité dans chaque secteur (Figure 2.7).

La procédure de reconnaissance consiste alors à mesurer le degré de similitude entre les matrices issues des modèles et celles de l'objet à reconnaître. Cette méthode a été appliquée sur une base de caractères mono fonte, dans différentes orientations. L'auteur ne précise pas la qualité des données traitées mais annonce de bons taux de reconnaissance. Notons qu'une étude menée dans notre

laboratoire par L. Lefrère [Lefrère 1993] montre la grande sensibilité au bruit de cette approche, essentiellement à cause de la grande variabilité sur la détermination du cercle de rayon maximal.

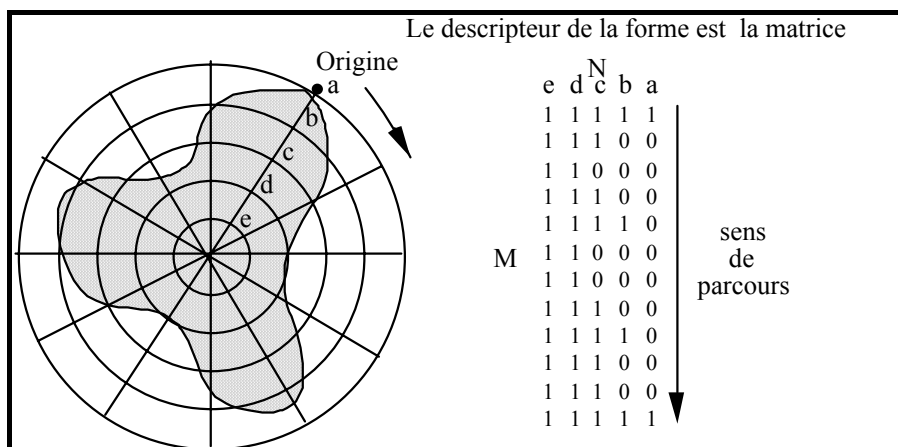


Figure 2.7 : Construction du descripteur circulaire de la forme selon Taza [Taza 1989]

La méthode proposée dans [Kita 1992] sous le nom de « **CCD** » (**C**oncentrique **C**irculaire **D**escription) repose sur le même ensemble de cercles concentriques de rayons proportionnels à la forme. L'auteur construit un vecteur caractéristique à partir de deux primitives : le nombre d'intersections d'un cercle avec la forme et la longueur totale de ces intersections. Dans cet article, une comparaison entre ce modèle et les invariants de Hu est proposée. Elle annonce la supériorité de l'approche de Kita.

Dans notre laboratoire, une méthode assez voisine est proposée dans [Lefrère 1993] pour la reconnaissance de caractères sur des cartes géographiques. Elle utilise des listes de valeurs angulaires issues du même ensemble de cercles concentriques. Une liste est obtenue en parcourant un cercle dans un sens donné. Lors de ce parcours, les écarts angulaires existant entre les points d'entrées et de sorties dans la forme sont mesurés et stockés dans la liste des valeurs angulaires. En fin de parcours, afin de rendre le problème invariant à la rotation, une permutation circulaire de la liste est effectuée en positionnant le secteur angulaire le plus important en tête (Figure 2.8). Cette méthode permet l'obtention de bons résultats pour une application traitant de la conversion de cartes. Nous l'avons reprise dans un but de comparaison. Nous verrons donc les résultats qu'elle permet d'obtenir dans la partie concernant la comparaison des résultats entre différents modèles que nous utilisons (2.4).

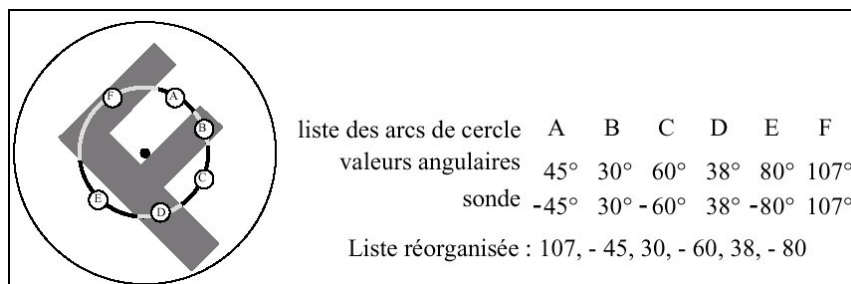


Figure 2.8 : Construction des listes caractéristiques selon [Lefrère 1993]

### 2.2.3.8. Synthèse des modèles extraits d'images binaires

Les modèles que nous avons présentés dans cette section peuvent être catégorisés en trois principaux types :

- ceux qui sont directement basés sur le *bitmap* de la forme, qui présentent l'intérêt d'être simples à extraire, mais pour lesquels l'invariance à la rotation semble difficile à obtenir si ce n'est par un apprentissage de très nombreuses formes.
- Les moments invariants : la plupart des modèles à base de moments présentés dans la partie précédente étaient difficiles à mettre en œuvre dans le cas d'images en niveaux de gris, malgré une théorie séduisante. Nous avons ici vu l'illustration de leur utilisation pour des images binaires au travers de quelques résultats quantitatifs. De ces résultats, nous retirons essentiellement la supériorité reconnue des modèles basés sur les moments de Zernike et leurs dérivés. Force est de constater que toute équipe s'attaquant au problème de la reconnaissance de forme invariante s'est un jour intéressée à ces moments de Zernike, au moins pour comparer leur approche à cette référence établie. Pour conclure sur ces approches à base de moments notons enfin que si la plupart des travaux cités dans cet état de l'art sont anciens, de nombreux chercheurs continuent à travailler dans cette voie. Néanmoins, leurs travaux ne consistent plus à développer de nouveaux modèles mais se déclinent suivant 3 axes :
  - Optimiser les approches existantes du point de vue de leur complexité. C'est par exemple le cas dans [Wu 2001], où l'image est préalablement décomposée en *quad-tree*, avant le calcul des moments ou dans [Sossa-Azuela 2001] qui utilise des opérations morphologiques pour décomposer les formes en une partition de disques.
  - Généraliser les différentes approches à base de moments pour en déduire une approche générale et prouver différentes propriétés d'un point de vue théorique. C'est par exemple le cas dans [Flusser 2000].
  - Dédier plus particulièrement les moments existants à une application donnée, en fonction des contraintes que celle-ci impose, comme par exemple dans [Balslev 2000]
- Notons enfin, l'existence d'approches structurelles à base d'occlusions qui consistent à représenter les formes par des graphes de composantes connexes de la couleur du fond. C'est le cas dans le système développé par l'équipe de Yu [Yu 1997] pour *Lucent Technologies*, qui vise à reconnaître une large gamme de schémas d'ingénierie tels que les diagrammes de circuits logiques, les schémas électriques. Nous détaillerons cette contribution dans la seconde grande partie puisqu'elle fait partie des rares systèmes complets publiés. Une approche similaire a été développée dans notre équipe pour la reconnaissance de symboles [Adam 1998][Adam 2000a]. Elle consiste à

construire un graphe dont les nœuds sont des occlusions préalablement classifiées et dont les arcs sont des relations de proximité entre ces occlusions (ce qui la rend utilisable quelle que soit l'orientation et en présence de connexions avec les éléments graphiques). Les résultats sont très bons, mais l'approche est très spécifique aux symboles France Telecom, présentant des compositions d'occlusions. Nous présentons en détail cette approche ainsi que de premières pistes de généralisation en Annexe B.

## 2.2.4. Les modèles issus des contours de la forme

Jusqu'alors, l'ensemble des modèles que nous avons présenté était extrait du bitmap de la forme. Dans cette partie, ce n'est plus le cas puisque nous considérons les modèles issus uniquement des contours de la forme. Pour pouvoir extraire le modèle il est donc nécessaire d'avoir recours à des pré-traitements. Le contour d'une forme est ici défini comme étant une courbe passant par le centre de tous les pixels 4-connexes au fond de l'image. Nous présentons ci-dessous 4 types de modèles issus des contours d'une forme.

### 2.2.4.1. Les profils

En suivant la définition ci-dessus, un contour peut être assimilé à un couple de fonctions discrètes à une seule variable ( $x$  ou  $y$ ) correspondant chacune à un demi-contour de la forme (Figure 2.9), chaque demi contour est délimité par les points extrêmes de la forme (haut / bas ou gauche / droite)

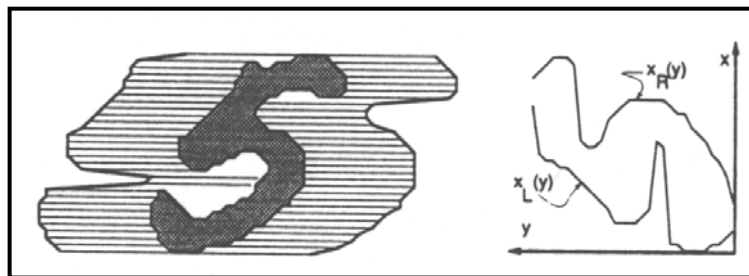


Figure 2.9 : Illustration des profils ( $x_L$  est le profil gauche,  $x_R$  le profil droit ([Trier 1996])

A partir de ces profils, plusieurs approches sont possibles. L'intégralité du profil, la dérivée des fonctions, la différence entre profil droit et profil gauche ou des valeurs discrètes caractéristiques comme les pics par exemple peuvent être utilisées en tant que modèle [Kimura 1991]. Notons aussi que la même approche avec profil haut et profil bas peut évidemment être exploitée. Néanmoins, quel que soit le choix effectué, il semble impossible d'aboutir à une invariance en rotation du modèle.

### 2.2.4.2. Le découpage en zones

Comme dans le cas des images en niveaux de gris ou binaires, certains auteurs font parfois appel à des techniques de découpage des contours en zones pour modéliser les formes. C'est par exemple le cas dans [Kimura 1991]. Dans cette contribution, l'orientation principale du contour est évaluée dans chaque zone et utilisée comme modèle de la forme, comme le montre la Figure 2.10.

Des approches similaires sont utilisées dans [Takahashi 1991], qui exploite en plus l'information point de rupture de courbe ou encore dans [Cao 1994] qui exploite le flou pour rendre l'extraction du modèle moins sensible à de petites variations des formes. Néanmoins, une fois de plus, quelle que soit l'approche adoptée parmi ces trois contributions, aucune ne permet la description de la forme indépendamment de l'orientation de celle-ci, ce qui la rend inutilisable dans le cadre de notre étude.

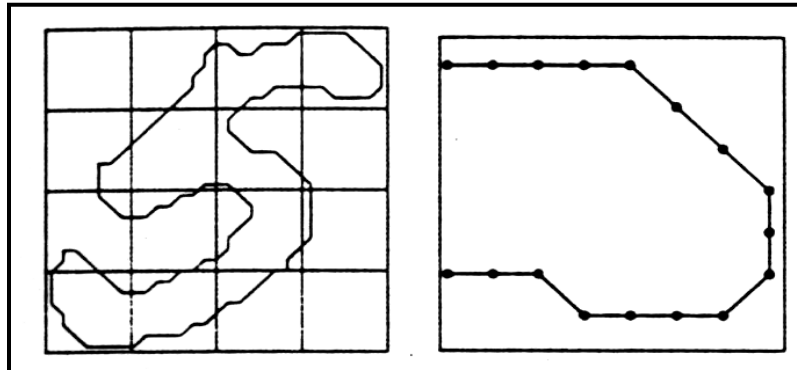


Figure 2.10 : Principes de la méthode de Kimura [Kimura 1991][Trier 1996]

### 2.2.4.3. Les approximations paramétriques

Pour modéliser les formes à partir de leurs contours, certains auteurs basent leur approche sur une estimation paramétrique de la courbe joignant les points de forte courbure. C'est par exemple le cas dans [Sekita 1988], où des *splines* sont utilisées pour approximer le contour entre des points d'arrêt. Le modèle est alors constitué à la fois par ces points d'arrêt, correspondant à des fortes courbures, et par les paramètres de la fonction *spline* décrivant le contour. Ce type de technique est également utilisé dans [Taxt 1990], qui y ajoute un lissage de la *spline* obtenue pour en réduire la sensibilité au bruit. Là encore, si l'invariance à la translation est évidente, et si l'invariance aux changements d'échelle peut être obtenue par normalisation, il semble difficile de rendre le modèle invariant aux changements d'orientation.

### 2.2.4.4. Les descripteurs de Fourier et leurs dérivés

Avec les moments de Zernike précédemment évoqués, le modèle basé sur les descripteurs de Fourier est probablement le plus utilisé dans la littérature. Il a pour intérêt principal d'être invariant à certaines transformations classiques telles que la rotation et le changement d'échelle, sous certaines contraintes. Le contour d'une forme est dans ce cadre considéré comme une courbe fermée de longueur  $L$ . En calculant une fonction cumulée  $\Phi$  des changements de direction angulaire le long de cette courbe (Figure 2.11), il est alors possible de construire une fonction périodique qui peut être décomposée en série de Fourier. Cette décomposition est à la base des descripteurs de Fourier [Zahn 1972].

La fonction cumulative des angles est définie telle que :

$$\phi(t) = \theta(t) - \theta(0) \tag{54}$$

Elle vérifie alors :

$$\phi(0) = 0 \quad \text{et} \quad \phi(L) = 2\pi = 0 \quad (55)$$

Une normalisation de cette fonction périodique sur l'intervalle  $[0 ; L]$  permet de construire une fonction  $\phi^*$  définie par :

$$\phi^*(t) = \phi\left(\frac{tL}{2\pi}\right) \quad (56)$$

Cette fonction étant périodique sur l'intervalle  $[0 ; 2\pi]$ , elle peut être décomposée en série de Fourier :

$$\phi^*(t) = \sum_{k \in \mathbb{Z}} C_k e^{ikt} \quad \text{avec} \quad C_k = \frac{1}{2\pi} \int_0^{2\pi} \phi^*(t) e^{-ikt} dt \quad (57)$$

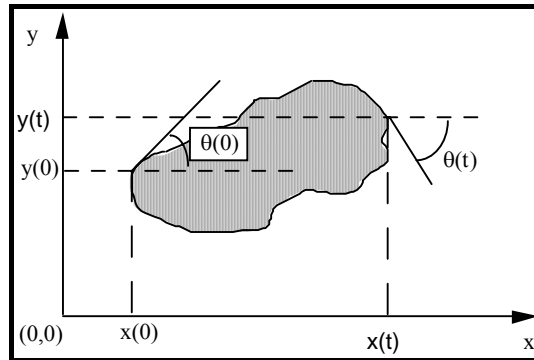


Figure 2.11 : Principes de construction des descripteurs de Fourier

Les descripteurs de Fourier peuvent alors être définis par le module des coefficients  $C_k$ . Notons que [Persoon 1977] souligne les inconvénients de cette méthode, en établissant, d'une part, que les suites ne représentent pas forcément des courbes fermées et, d'autre part, que les occlusions des formes ne sont pas décrites. Une approche voisine est présentée dans [Granlud 1972]. Les descripteurs de Fourier  $y$  sont définis différemment. Le contour fermé de périmètre  $L$  est représenté dans l'espace cartésien par les points de coordonnées  $x(l)$  et  $y(l)$  où  $l$  décrit la longueur de l'arc par rapport à un point origine ( $l$  varie de  $0$  à  $L$ ). Ces fonctions étant périodiques de période  $L$ , la fonction complexe décrivant le contour peut alors être décomposée en série de Fourier :

$$x(m) = \sum_{n=-\infty}^{+\infty} a(n) \cdot e^{jn\omega_0 m} \quad \text{et} \quad y(m) = \sum_{n=-\infty}^{+\infty} b(n) \cdot e^{jn\omega_0 m} \quad \text{avec} \quad \omega_0 = 2\pi/L \quad (58)$$

Les coefficients complexes de Fourier sont alors dérivés par :

$$a(n) = \frac{1}{L-1} \sum_{m=1}^{L-1} x(m) e^{-jn\omega_0 m} \quad \text{et} \quad b(n) = \frac{1}{L-1} \sum_{m=1}^{L-1} y(m) e^{-jn\omega_0 m} \quad (59)$$

Pour obtenir l'invariance à la rotation, les auteurs [Zahn 1972], [Badreldin 1980], [Wu 1986] utilisent les valeurs  $r(n)$  suivantes :

$$r(n) = \sqrt{|a(n)|^2 + |b(n)|^2} \quad (60)$$

Par une simple normalisation, des coefficients ne variant pas en fonction de l'échelle de la forme peuvent alors être calculés :

$$s(n) = \frac{r(n)}{r(1)} \quad (61)$$



De très nombreux auteurs ont adopté ce modèle, surtout lorsqu'il existe des contraintes d'invariance. Citons par exemple [Shridhar 1984] qui obtient, par l'intermédiaire de ce modèle, un score de 99% de reconnaissance. Néanmoins, comme toujours, ces résultats peuvent difficilement permettre de tirer des conclusions puisqu'ils ne sont pas comparés à d'autres approches et que les bases ne sont pas précisément décrites.

Outre ces descripteurs de Fourier, une autre approximation du contour d'une forme peut être effectuée avec les descripteurs elliptiques de Fourier [Kuhl 1982][Lin 1987][Taxt 1990]. En posant  $t = 2\pi m/L$ , le contour peut être décrit sous une forme matricielle par un développement en série de Fourier :

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} a_0 \\ d_0 \end{bmatrix} + \sum_{k=1}^{+\infty} \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \begin{bmatrix} \cos kt \\ \sin kt \end{bmatrix} \quad (62)$$

Avec :

$$\begin{aligned} a_0 &= \frac{1}{2\pi} \int_0^{2\pi} x(t) dt & a_k &= \frac{1}{\pi} \int_0^{2\pi} x(t) \cos(kt) dt & c_k &= \frac{1}{\pi} \int_0^{2\pi} y(t) \cos(kt) dt \\ d_0 &= \frac{1}{2\pi} \int_0^{2\pi} y(t) dt & b_k &= \frac{1}{\pi} \int_0^{2\pi} x(t) \sin(kt) dt & d_k &= \frac{1}{\pi} \int_0^{2\pi} y(t) \sin(kt) dt \end{aligned} \quad (63)$$

Avec une telle écriture,  $a_0$  et  $b_0$  définissent le centre géométrique du contour alors que les  $x_k(t)$  et les  $y_k(t)$  donnés par (64) définissent la  $k^{\text{ième}}$  ellipse, dont  $E_k$  est la matrice.

$$\begin{bmatrix} x_k(t) \\ y_k(t) \end{bmatrix} = \begin{bmatrix} a_k & b_k \\ c_k & d_k \end{bmatrix} \begin{bmatrix} \cos(kt) \\ \sin(kt) \end{bmatrix} = E_k \begin{bmatrix} \cos(kt) \\ \sin(kt) \end{bmatrix} = r_k \begin{bmatrix} A_k & 0 \\ 0 & B_k \end{bmatrix} P_k \begin{bmatrix} \cos(kt) \\ \sin(kt) \end{bmatrix} \quad (64)$$

La Figure 2.12 montre une illustration des ellipses. Les valeurs  $A_k$  et  $B_k$  représentent les longueurs des demi-axes de l'ellipse. Les valeurs dans la matrice  $r_k$  et  $P_k$  caractérisent quant à elles l'orientation et le point de départ de l'ellipse. Ces valeurs  $r_k$  et  $P_k$  sont données par :

$$r_k = \begin{bmatrix} \cos \theta_k & -\sin \theta_k \\ \sin \theta_k & \cos \theta_k \end{bmatrix} \quad \text{et} \quad P_k = \begin{bmatrix} \cos \Psi_k & -\sin \Psi_k \\ \sin \Psi_k & \cos \Psi_k \end{bmatrix} \quad (65)$$

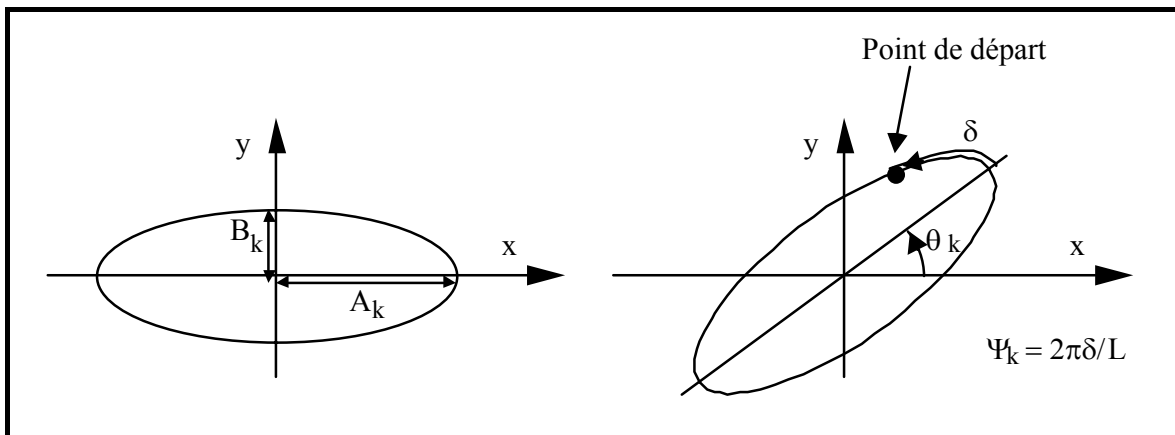


Figure 2.12 : Illustration intuitive des ellipses dans les descripteurs elliptiques de Fourier

Avec une telle approche, la forme est représentée par un ensemble d'ellipses. Chacune d'elles est définie par une orientation et une position par rapport à la précédente. Les descripteurs constituant le modèle décrivent donc à la fois les relations inter-ellipses et intra-ellipses, ce qui permet de les rendre invariants aux transformations que sont les rotations et les translations. La formulation mathématique de ces descripteurs est donnée par l'équation (66).

$$\begin{aligned} I_k &= a_k^2 + b_k^2 + c_k^2 + d_k^2 \\ J_k &= a_k d_k - b_k c_k \\ K_{jk} &= \text{trace}(E_k E_k^T E_j E_j^T) \end{aligned} \quad (66)$$

L'obtention de l'invariance au changement d'échelle se fait classiquement par normalisation, en utilisant la première ellipse, c'est à dire en divisant chaque descripteur issu de (66) par :

$$D = \sqrt{a_1^2 + b_1^2 + c_1^2 + d_1^2} \quad (67)$$

Une étude comparative très complète entre les descripteurs elliptiques de Fourier de Lin [Lin 1987], les descripteurs elliptiques de Fourier de Kuhl [Kuhl 1982] et les descripteurs de Fourier de Zahn [Zahn 1972] est proposée dans [Taxt 1990]. La comparaison est effectuée sur une base de caractères comprenant une centaine d'exemples par classe. Cette étude montre la supériorité des descripteurs elliptiques, que l'orientation de la forme soit connue au préalable ou non. La différence en faveur des descripteurs elliptiques est d'environ 4 points. Ces études sont corroborées par les travaux de Weisteen [Weisteen 1992].

#### **2.2.4.5. Des invariants issus des contours et d'une transformée en ondelettes**

Nous avons vu en 2.2.2.5 que les décompositions en ondelettes étaient de plus en plus utilisées pour la reconnaissance de formes. Nous avons donné deux exemples de travaux se basant sur une telle approche en partant de l'image des formes. Ici, nous présentons une étude récente d'une équipe canadienne [Khalil 2000] qui exploite la théorie des ondelettes appliquée aux contours des formes. A la suite de l'extraction des contours, 2 principales étapes peuvent être distinguées :

- La première étape consiste à représenter l'ensemble des points du contour dans un diagramme  $\Theta$ - $S$ . La variable  $\Theta$  représente l'angle formé par la tangente au contour en un point donné par rapport à une droite de référence. Cette variable est calculée dans une fenêtre  $W$  (équation (68)).

$$\Theta_i = \tan^{-1} \frac{y_i - y_{i-w}}{x_i - x_{i-w}} \quad i = 0, 1, \dots, N-1 \quad (68)$$

$W$  est la taille de la fenêtre.  $x_i$  et  $y_i$  sont les coordonnées des  $N$  points de contour. Notons qu'un offset est ajouté à la valeur de  $\Theta$  pour éviter les discontinuités. La variable  $S$  du diagramme  $\Theta$ - $S$  représente l'abscisse curviligne du point pris en compte. Dans le diagramme, la nouvelle représentation devient invariante à la rotation. Les invariances à la translation et au changement

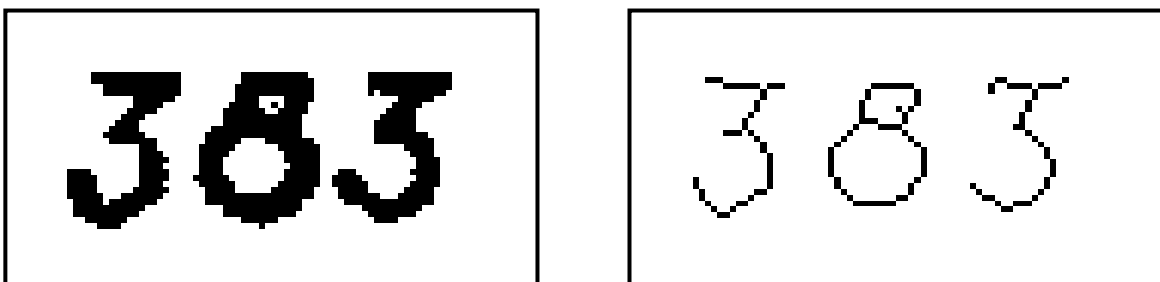
d'échelle sont respectivement obtenues par le choix d'un point de départ de parcours du contour stable, et par normalisation du nombre de points sur le contour à 256.

- Le diagramme ainsi obtenu est ensuite analysé par une transformée en ondelettes. Nous ne détaillons pas ici cette transformation, que nous avons déjà présentée. L'analyse locale que permet la théorie des ondelettes aboutit à l'extraction des points « singuliers », correspondant à de fortes ruptures dans les contours. Ces sont alors les distances entre ces points qui sont utilisées comme représentation des formes.

En associant à ces invariants extraits un classifieur de type perceptron multi-couches, les auteurs obtiennent de très bons résultats dans le cadre de leur application qui consiste à reconnaître des images d'outils. Si la théorie est séduisante, les résultats présentés ne sont pas suffisants pour valider l'approche. Notons tout de même que la preuve pratique de l'insensibilité au bruit est montrée dans l'article.

### **2.2.5. Les modèles issus du squelette des formes**

Le squelette d'une forme peut être défini comme étant l'axe central - d'épaisseur unitaire - de la (éventuellement les) composante(s) connexe(s) constituant la forme. De très nombreux algorithmes ont été développés pour obtenir le squelette d'une image. Des études bibliographiques, menées dans [Tombre 1996][Adam 1998] ou [Mullot 2000], montrent que chaque algorithme a des avantages, soit en terme de précision ou en terme de rapidité de calcul. Elles montrent aussi que, quel que soit le choix effectué, le résultat restera très sensible aux bruits, et que les points de jonction constituent un problème majeur. La Figure 2.13 donne une illustration d'une telle sensibilité.



*Figure 2.13 : Illustration du squelette (à droite) d'une image de caractères*

Ce squelette peut directement être utilisé en tant qu'image pour la reconnaissance, en utilisant les approches présentées en 2.2.3.1. De plus, il est aussi possible d'en extraire des modèles différents, qui exploitent l'information mise en avant par le squelette.

### **2.2.5.1. Les descriptions structurelles basées sur le squelette**

Jusqu'alors, nous avons essentiellement évoqué des modèles numériques (souvent qualifiés de statistiques) de forme. Les caractères ou les symboles sont alors représentés à partir de vecteurs issus de mesures ou de transformations mathématiques de l'image. L'information manipulée n'est alors pas très symbolique. Une autre alternative, généralement adoptée pour la reconnaissance de symbole consiste à utiliser une représentation structurelle de la forme. Des structures de données abstraites (chaînes, graphes, arbres, listes) sont alors manipulées. Dans ce type d'approches, la reconnaissance de la forme est généralement obtenue par comparaison de sa représentation symbolique par rapport à celle des modèles prédéfinis. Des techniques de recherche de sous graphes communs et d'appariement de graphes sont alors utilisées. Diverses approches ont été développées dans ce cadre (isomorphismes, exacts ou inexacts, de sous graphe, relaxation de contraintes, recherche de cliques maximales, algorithmes génétiques ...). Le lecteur trouvera une bonne synthèse de tels outils dans [Ah Soon 1998][Chhabra 1998][Dosch 2000] ou [Lladós 2001]. Dans le cas de l'utilisation du squelette, ce sont les vecteurs et les arcs de cercles qui forment les nœuds du graphe. Les arcs du graphe représentent quant à eux des relations de connexion, de distances, d'orientation relative entre ces vecteurs et arcs. De très nombreux systèmes de reconnaissance de symboles ont été développés sur ce principe, citons par exemple [Messmer 1996] et [Ah Soon 1998]. L'avantage principal de ces approches réside dans la possibilité de gérer des symboles connectés. Toutefois, ce genre d'approches ne semble pas, au vu de la littérature, suffisamment robuste pour reconnaître les caractères isolés

### **2.2.5.2. Les Primitives discrètes**

Sans avoir recours à un graphe décrivant complètement le squelette, il est aussi possible d'extraire de celui-ci un certain nombre de valeurs discrètes caractéristiques de caractères. Parmi celles-ci citons celles exploitées par Ramesh [Ramesh 1989], Kundu [Kundu 1989] et [Heutte 1998] : le nombre de boucles ; les nombres de jonctions, en « T » ou en « X » ; le nombre de points terminaux en général ou dans chacune des directions et le ratio hauteur / largeur du squelette. Si ces primitives sont très faciles à extraire, il est globalement reconnu qu'elles sont peu robustes à de petites variations des formes.

### **2.2.5.3. Les descripteurs de Fourier**

Comme pour les contours, il est possible d'utiliser les descripteurs de Fourier pour décrire le squelette d'une forme. Si la théorie impose en principe une périodicité des fonctions, qui se traduit concrètement par la fermeture des lignes, des études ont montré qu'il était néanmoins envisageable d'appliquer de telles approches au squelette. Dans [Taxt 1994], des descripteurs ayant la propriété d'invariance à la rotation et au changement d'échelle sont proposés, en appliquant au préalable une analyse du nombre de boucles, du nombre de nœuds et du nombre de points terminaux. En effet, il est prouvé dans cet article que dans le cas de graphes contenant des intersections et des boucles, tous les

descripteurs sont non nuls. Si le squelette est sans boucle, et que seuls 2 points terminaux sont présents, certains descripteurs prendront en revanche la valeur nulle.

### 2.2.6. Synthèse de l'étude bibliographique

Dans le tableau qui suit, nous dressons un bilan des différentes approches que nous avons abordées dans le cadre de cette étude bibliographique. Pour chacun des cas, nous précisons dans ce tableau l'adéquation du modèle par rapport à notre problématique, à savoir si le modèle prend en compte :

- Une possible multi-orientation des formes (symbole ♣)
- Une possible multi-échelle des formes (♦)
- Un possible traitement des formes connectées sans que le temps de calcul soit prohibitif (♠)
- Un possible calcul de l'orientation (●)

<i>Image en niveaux de gris</i>	<i>Image binaire</i>	<i>Contours</i>	<i>Squelette</i>
Image elle-même ♦	Image elle-même ♦	Profils ♦	Graphes ♦ ♠
Transformées Unitaires ♦	Transformées unitaires ♦	Découpages en zones ♦	Attributs discrets ♣ ♦
Moments de Hu ♣ ♦	Projections ♦	Approximations ♦	Descripteurs Fourier ♣ ♦
Moments de Bamieh ♣ ♦	Moments de Hu ♣ ♦	Descripteurs Fourier ♣ ♦	
Moments de Zernike ♣ ♦	Moments de Bamieh ♣ ♦	Descripteurs Elliptiques	
Moments de Pseudo Zernike ♣ ♦	Moments de Zernike ♣ ♦ ●	de Fourier ♣ ♦	
Ondelettes ♣ ♦	Structurel ♣ ♦	Ondelettes ♣ ♦	
	Primitives circulaires ♣ ♦		
	Ondelettes ♣ ♦		

*Tableau 2.2 : Synthèse des modèles de la littérature*

Au vu du grand nombre de modèles invariants proposés dans la littérature, il pourrait être argumenté qu'il n'est pas nécessaire d'en développer de nouveaux. La priorité pourrait être donnée, d'une part à leur optimisation et d'autre part à une meilleure exploitation des résultats par des traitements de plus haut niveau. C'est le raisonnement qu'ont adopté la plupart des équipes (hormis les récents travaux sur les ondelettes qui ouvrent de très intéressantes perspectives) puisqu'une analyse des références montre que les contributions sont actuellement moins nombreuses qu'au début des années 1990. Cela n'est d'ailleurs pas limité aux outils de reconnaissance de texte mais tend à se généraliser à tous types d'outils.

Pourtant, les études comparatives, en particulier celle menée dans [Belkasim 1991] montrent que les différents modèles sont encore loin d'être parfaits quant aux performances de reconnaissance. De plus, cette analyse de l'état de l'art met également en évidence la difficulté de la mise en place d'une

stratégie générique pour l'analyse de caractères/symboles connectés, ce qui constitue une contrainte majeure pour notre application. C'est pourquoi nous présentons ci-après, un nouveau modèle de représentation des formes, le modèle de Fourier Mellin qui permet d'apporter des solutions à ces problèmes.

## 2.3. Un nouveau modèle de représentation des formes basé sur la Transformée de Fourier Mellin (TFM)

Ce chapitre est dédié à la présentation de ce qui constitue probablement la plus importante contribution dans le cadre de cette thèse. Nous y présentons un modèle de représentation des formes jusqu'alors inexploité dans le domaine de l'interprétation de documents techniques. Celui-ci est basé sur la Transformation de Fourier Mellin (TFM). Il permet la description d'une forme par un ensemble de descripteurs qui sont invariants quelles que soient la position, l'échelle et l'orientation de la forme analysée ainsi que la gestion des formes connectées. Dans ce chapitre, nous rappelons d'abord quelques concepts mathématiques nécessaires à une bonne assimilation de cette transformée. Puis, nous en présentons la définition, en la comparant à la formulation du prolongement analytique de la TFM définie dans [Ghorbel 1994]. Ensuite, nous montrons comment appliquer cette transformée continue au signal bidimensionnel discret caractéristique d'une image. Cela nous permet de décrire l'algorithmique d'extraction de ce modèle dans le cadre de notre étude sur les documents techniques. Enfin, nous verrons quelques illustrations des différentes propriétés de notre modèle. Les résultats quantitatifs obtenus feront quant à eux l'objet du chapitre suivant.

### 2.3.1. Quelques rappels

D'un point de vue mathématique, la Transformée de Fourier Mellin peut être analysée de deux manières différentes. Le premier de ces points de vue est celui de la théorie des groupes et de l'analyse harmonique. Dans ce contexte, la TFM se présente alors comme une transformée de Fourier généralisée au groupe des similitudes planes vectorielles, c'est à dire au produit direct du groupe des rotations par le groupe des dilatations. Le second point de vue est plus intuitif ; il consiste à voir la TFM comme l'association de deux transformations souvent utilisées en analyse des signaux et des images : les Fonctions Harmoniques Circulaires (FHC) et de la Transformée de Mellin (T.M.) radiale. C'est ce point de vue que nous avons choisi dans le cadre de ce mémoire. C'est pourquoi nous rappelons dans cette section les définitions respectives de ces deux transformations, afin d'aboutir en 2.3.2 à une définition intuitive de la TFM. Le lecteur intéressé par une construction plus orientée « théorie de groupes » de la TFM peut se reporter aux références [Pavel 1989], [Lenz 1989], [Kanatani 1990], [Mundy 1993], [Ghorbel 1994] et [Derrode 1999] pour une présentation très complète.

#### 2.3.1.1. Les Fonctions Harmoniques Circulaires (FHC)

Les Fonctions Harmoniques Circulaires (FHC) sont utilisées depuis longtemps en traitement d'images pour leur invariance à la rotation. D'un point de vue définition, si  $f$  est une fonction bidimensionnelle représentant l'image en coordonnées cartésiennes, elle doit d'abord être exprimée en coordonnées polaires, sous la forme  $g(r, \theta)$ , en appliquant le changement de variables :

$$r = \sqrt{x^2 + y^2} \quad \text{et} \quad \theta = \arctan \frac{y}{x} \quad (69)$$

Les FHC de cette fonction  $g$  sont alors les coefficients de Fourier de chacun des cercles de rayon  $r$  de l'image. Ils prennent la forme :

$$g_q(r) = \frac{1}{2\pi} \int_0^{2\pi} g(r, \theta) e^{-iq\theta} d\theta, \quad q \in \mathbb{Z} \quad (70)$$

Les valeurs de ces fonctions sont dépendantes du choix de l'origine des coordonnées. Généralement, le centre de gravité des formes (aussi appelé centre de masse) est choisi, afin d'aboutir à une invariance en translation. Cette transformée de Fourier particulière respecte les propriétés d'une transformée de Fourier classique, à savoir :

- La complétude : la représentation image originale peut être reconstruite à l'aide des séries de Fourier, en utilisant un ensemble infini de ses FHC et l'équation :

$$g(r, \theta) = \sum_{q \in \mathbb{Z}} g_q(r) e^{iq\theta} \quad (71)$$

- L'application possible du théorème du décalage ou théorème du retard :

$$\text{si } g'(r, \theta) = g(r, \beta + \theta) \quad \text{alors} \quad g'_q(r) = e^{iq\beta} g_q(r) \quad \forall q \in \mathbb{Z} \quad (72)$$

Nous verrons que cette dernière propriété est fondamentale quant à l'obtention de l'invariance à la rotation de la transformée de Fourier Mellin.

### 2.3.1.2. La Transformée de Mellin (T.M.)

La Transformée de Mellin (T.M.) a, elle aussi, été largement utilisée dans le domaine du traitement d'images, essentiellement pour le traitement d'images radar [Inggs 1995]. Elle est, comme les Fonctions Harmoniques Circulaires, une voisine de la transformée de Fourier, mais appliquée au groupe des dilatations. Elle est applicable à des fonctions causales respectant :

$$\int_0^{\infty} |f(r)| \frac{dr}{r} < \infty \quad (73)$$

Si  $f$  vérifie cette condition, et appartient donc à l'espace vectoriel correspondant, la TM est définie par :

$$\forall \nu \in \mathbb{R}, \quad M_f(\nu) = \int_0^{\infty} f(r) r^{-i\nu} \frac{dr}{r} \Leftrightarrow M_f(\nu) = \int_0^{\infty} f(r) r^{-i\nu-1} dr \quad (74)$$

Cette transformée vérifie la propriété de complétude, sous réserve de l'existence de l'intégrale de droite dans l'équation suivante :

$$\forall r \in \mathbb{R}_+^*, \quad f(r) = \int_{-\infty}^{+\infty} M_f(\nu) r^{i\nu} d\nu \quad (75)$$



Le théorème du retard s'applique également, mais cette fois pour des dilatations du signal, c'est à dire que :

$$\text{si } g(r) = f(\alpha r) \text{ alors } M_g(\nu) = \alpha^{i\nu} M_f(\nu) \quad (76)$$

Notons ici que pour des raisons d'approximations, le changement de variables  $t=ln(r)$  est souvent adopté, ce qui permet la définition suivante de la TM :

$$\forall \nu \in \mathbb{R}, \quad M_f(\nu) = \int_{-\infty}^{+\infty} f(e^t) e^{-i\nu t} dt = 2\pi F_g(\nu) \quad (77)$$

Dans cette expression  $g(t) = f(e^t)$  et  $F_g$  est la transformée de Fourier de la fonction  $g$ . La TM est donc en réalité une Transformée de Fourier appliquée à un signal échantillonné à pas exponentiel.

### **2.3.2. Définition de la Transformée de Fourier Mellin (TFM)**

Soit  $f(r, \theta)$  une fonction réelle et positive (la forme) définie en coordonnées polaires. La TFM de cette fonction est définie par la relation suivante :

$$M_f(q, s) = \int_{r=0}^{\infty} \int_{\theta=0}^{2\pi} r^{s-1} \exp(-iq\theta) f(r, \theta) dr d\theta \quad \text{avec } q \in \mathbf{Z}, s = \sigma + i\nu \in \mathbf{C} \quad (78)$$

Dans cette expression,  $i$  désigne l'unité imaginaire. La TFM est donc la combinaison des Fonctions Harmoniques Circulaires d'une fonction  $2\pi$ -périodique pour la variable angulaire  $\theta$ , et de la transformée de Mellin pour la variable radiale  $r$ . La TFM existe pour les valeurs de  $(q, s)$  pour lesquelles la double intégrale (78) est définie. En pratique, pour une fonction  $f(r, \theta)$  positive donnée, la convergence est obtenue pour tout  $q$ , mais pas pour tout  $s = \sigma + i\nu$ . En effet, si  $f(r, \theta)$  se comporte au voisinage de l'origine comme  $Kr^\lambda$ , alors la TFM sera définie à condition que  $\sigma > -\lambda$ . En général, le comportement de la forme au voisinage de l'origine correspond à des valeurs de  $\lambda$  proches de 0, ce qui implique souvent l'emploi d'une valeur de  $\sigma$  constante et comprise dans l'intervalle  $[0.5 ; 1.5]$ . La TFM est alors calculée pour  $s$  décrivant le contour de Bromwich  $]\sigma - i\infty, \sigma + i\infty[$ .

Notons qu'il existe une définition plus restreinte de la TFM, correspondant à  $\sigma=0$ , décrite dans [Ghorbel 1994] et [Derrode 1999]. Cette autre définition trouve son origine dans la théorie des groupes et l'analyse harmonique (cf. 2.3.1). En effet, en élargissant la notion de transformée de Fourier aux groupes de transformations topologiques, compacts ou localement compacts, et commutatifs, Ghorbel [Ghorbel 1994] a montré que la transformée de Fourier sur le groupe des similitudes conduisait à cette autre définition de la TFM, également adoptée dans [Wood 1996]. Afin de pallier les problèmes de divergence de (78) pour  $\sigma=0$ , l'auteur a proposé la définition d'un prolongement analytique de la TFM (TFMA), qui revient tout simplement à l'écriture de (78) avec  $\sigma > 0$ , et assure ainsi la convergence de la TFM, en particulier pour les images numériques. Pour résumer, la TFM avec  $\sigma=0$  est à la TFM de l'éq. (78) ce qu'est la transformée de Fourier à la transformée de Laplace d'un signal continu.

En utilisant cette définition, la fonction  $f(r, \theta)$  peut être reconstruite à partir de la TFM, en utilisant les relations suivantes :

$$f(r, \theta) = \frac{1}{2\pi i} \int_{\sigma-i\infty}^{\sigma+i\infty} \sum_{q \in \mathbb{Z}} r^{-s} \exp(iq\theta) M_f(q, s) ds \quad (79)$$

donc

$$f(r, \theta) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \sum_{q \in \mathbb{Z}} r^{-\sigma-iv} \exp(iq\theta) M_f(q, s = \sigma + iv) dv \quad (80)$$

Cette relation définit la TFM inverse qui permet, à partir de l'expression de la TFM de  $f$ , de revenir au signal initial.

### 2.3.3. Des invariants issus de la transformée de Fourier Mellin

La construction de descripteurs de formes invariants par rotation et homothétie s'appuie sur l'application du théorème du retard, classique dans toute l'analyse de Fourier (au même titre que pour les *FHC* et la *TM* précédemment définis). Dans ce contexte, soit  $g(r, \theta) = f(\alpha r, \theta + \beta)$  une version modifiée en orientation (d'un terme angulaire  $\beta$ ) et en échelle (d'un facteur  $\alpha$ ) d'une fonction initiale  $f(r, \theta)$ . Il peut être facilement montré, par changement de variables, la relation suivante :

$$M_g(q, s) = \alpha^{-s} \exp(iq\beta) M_f(q, s) \quad (81)$$

En prenant le module des termes de l'équation (81), il est donc possible de déduire des descripteurs qui sont invariants aux changements de rotation de la forme mais pas aux changements d'échelle. Pour obtenir l'invariance au changement d'échelle, il est possible d'utiliser l'ensemble de descripteurs suivants :

$$I_f(q, s) = |M_f(q, s)| [M_f(0, \sigma)]^{-1} \quad (82)$$

Cet ensemble d'invariants fournit une représentation simple des formes. Cependant, il ne vérifie pas la propriété de complétude, c'est-à-dire qu'il n'existe pas de bijection entre les représentations duales d'une forme, à cause de la perte de l'information de phase. Dans [Ghorbel 1994], l'auteur propose un ensemble de primitives invariants aux changements d'échelle et d'orientation ; celui-ci repose sur l'expression suivante, généralisée ici à une valeur de  $\sigma$  quelconque :

$$I_f(q, s) = M_f(q, s) [M_f(0, \sigma)]^{-\frac{s}{\sigma}} [M_f(1, \sigma)]^{-q} |M_f(1, \sigma)|^q \quad (83)$$

Partant de ces invariants, il peut être montré facilement que :

$$\text{si } g(r, \theta) = f(\alpha r, \theta + \beta) \text{ alors } I_g(q, s) = I_f(q, s) \quad (84)$$

Les deux propriétés importantes de ces primitives sont alors les suivantes :

- i. *Complétude* : à partir des valeurs de  $\{I_f(q, s)\}$ ,  $M_f(0, \sigma)$  et  $M_f(1, \sigma)$ , il est possible de revenir à l'ensemble des coefficients de FM  $\{M_f(q, s)\}$ , et donc reconstruire  $f(r, \theta)$ , en utilisant la TFM inverse donnée par la relation (80).
- ii. *Convergence* : dans [Ghorbel 1994], il est prouvé, en supposant que  $\{M_f(q, s)\}$  est un ensemble de fonctions convergentes, qu'il existe  $x \in \mathbb{R}$ ,  $x > 1$  tel que :

$$\left( \int_{\mathbf{R}} \sum_{\mathbf{Z}} |I_f(q,s)|^x dv \right)^{1/x} < +\infty \quad (85)$$

Une conséquence importante de la propriété de convergence de cet ensemble d'invariants réside dans l'existence d'une métrique dans l'espace de représentation des formes. Nous verrons ultérieurement que l'existence d'une telle métrique est fondamentale dans le cadre du processus de reconnaissance de formes pour effectuer une classification.

### 2.3.4. Application à la structure bidimensionnelle discrète des images

De nombreuses difficultés surviennent lors de l'application de la TFM à des images numériques. La première vient de la nécessité d'échantillonner, suivant la variable (continue)  $v$ , les fonctions  $\{M_f(q,s)\}$ . Ce problème – ouvert – peut admettre une solution simple en échantillonnant régulièrement cette variable [Derrode 1999].

Le second problème réside dans le choix nécessaire d'un point invariant et représentatif de la forme comme Centre de Développement (CdD) de la TFM. Ce point permettra l'obtention de l'invariance à la translation et sera utilisé pour le calcul de la TFM. Comme pour tous les modèles de représentation présentés dans l'état de l'art, se pose alors le problème de la profondeur de l'image utilisée. En effet, si d'un point de vue théorique, la T.F.M. peut être appliquée à une image en niveaux de gris ( $f(r, \theta)$  prenant alors ses valeurs dans l'intervalle  $[0,255]$ ), il est alors difficile de borner les fenêtres sur lesquelles la TFM va être appliquée. Choisir un CdD invariant devient alors délicat. C'est pourquoi nous avons opté pour l'utilisation d'images préalablement binarisées pour l'extraction de notre modèle de Fourier Mellin. De ce fait, le choix du CdD est simplifié, et il devient naturel d'appliquer la TFM au centre de gravité de la forme. En effet, pour des formes binaires bien conditionnées (c'est à dire déconnectées de toute autre forme), le calcul des coordonnées du centre de gravité est facile et permet d'assurer que la forme est incluse dans un disque de rayon donné, argument qui sera d'un grand intérêt dans la suite. De plus, le centre de gravité d'une forme vérifie la propriété d'invariance à la rotation.

La troisième et principale difficulté est due à la structure bidimensionnelle, cartésienne et discrète de l'image numérique, sur laquelle il faut appliquer une transformation définie dans un repère polaire. Pour résoudre ce problème, deux options sont disponibles : l'interpolation de l'image en coordonnées polaires [Derrode 1999][Ghorbel 1994]; ou bien la convolution de l'image en coordonnées cartésiennes, au CdD d'une forme, avec un banc de filtres appropriés [Ravichandran 1995].

Avec la première option, la TFM est obtenue grâce à la relation (78) appliquée au CdD de la forme considérée. Cela suppose donc que l'image soit d'abord interpolée en coordonnées polaires pour couvrir la gamme entière de  $]0, r_{max}]$  pour la variable  $r$  et  $]0, 2\pi]$  pour la variable  $\theta$  ( $r_{max}$  étant le rayon maximal de la forme). Cette technique d'interpolation entraîne un coût de calcul qui peut être important, et peut aussi introduire de l'information redondante dans les données originales.

Avec la deuxième option, il n'est pas nécessaire d'interpoler la forme à partir de l'image. En effet, la version discrète de la TFM, calculée au CdD  $(k_0, l_0)$  peut être approchée de la façon suivante :

$$M_{f(k_0, l_0)}(q, \sigma + ip) \approx \sum_k \sum_{\substack{l \\ 1 \leq (k^2 + l^2) \leq r_{\max}^2}} h_{q,p}(k, l) f(k_0 - k, l_0 - l) \quad (86)$$

avec

$$h_{q,p}(k, l) = \frac{\exp\left\{i \left[ \frac{p}{2} \ln(k^2 + l^2) - q \arctan\left(\frac{l}{k}\right) \right]\right\}}{(k^2 + l^2)^{1 - \frac{\sigma}{2}}} \quad (87)$$

Cela signifie que la convolution de la forme avec un filtre 2-D de réponse impulsionnelle finie  $h_{q,p}(\dots)$  au CdD de la forme donne le coefficient de la TFM d'ordre  $(q, \sigma + ip)$ . Notons ici que tous les filtres (une illustration des valeurs est donnée en Figure 2.14), mis à part  $h_{0,0}(\dots)$  sont à valeurs complexes. Il peut être facilement démontré que l'ensemble des filtres  $h_{q,p}(\dots)$  est orthonormal en fonction du coefficient circulaire  $q$  mais pas en fonction du coefficient radial  $p$ , à cause du noyau de la transformée de Mellin, défini originellement pour une variable  $v$  continue. Une orthogonalisation complète des fonctions de base est néanmoins possible si des conditions spéciales sur la discrétisation de la variable  $v$ , et sur les valeurs de  $\sigma$  et  $r_{\max}$  sont respectées ; néanmoins, cette orthogonalisation n'a pas été effectuée dans ce travail puisque les résultats obtenus ne la rendaient pas nécessaire.

Dans le cadre de notre application, du fait de sa simplicité, nous avons opté pour la seconde option qui, de plus, offre l'avantage d'une implémentation parallélisable pour le calcul des coefficients de la TFM. Cet aspect pratique du calcul de la TFM fait d'ailleurs l'objet de recherches actuelles dans le laboratoire PSI. Nous y reviendrons dans le cadre de la conclusion générale de ce mémoire.

Dès lors que les coefficients de la TFM sont disponibles au CdD d'une forme donnée, l'Eq. (83) est utilisée pour générer des primitives invariantes aux changements d'échelle et d'orientation pour cette forme. Ces primitives sont toutes complexes, exception faite de  $I_f(0, \sigma)$  qui vaut 1 et de  $I_f(l, \sigma)$  qui est un réel pur. De plus, l'ensemble d'invariants vérifie la propriété de symétrie suivante :

$$I_f^*(q, s) = I_f(-q, s^*) \quad (88)$$

Cette propriété de symétrie hermitienne vis-à-vis des variables  $p$  et  $q$  est intéressante au regard du temps de calcul car elle permet de restreindre le nombre de primitives non redondantes à l'ensemble défini par :

$$\mathcal{A} = \left\{ (q, p) \mid (q = 0 ; 0 \leq p \leq P) \cup (1 \leq q \leq Q ; -P \leq p \leq P) \right\} \quad (89)$$

Notons ici qu'en utilisant l'approche telle que nous la présentons ci-dessus, et même si celle-ci pose un certain nombre de problèmes théoriques, la TFM discrète inverse, appliquée en coordonnées cartésiennes peut être approchée par la relation suivante :

$$f(k, l) \approx \frac{1}{2\pi} \sum_{q=-Q}^Q \sum_{p=-P}^P M_f(q, s = \sigma + ip) g_{q,p}(k, l) \quad , \quad (90)$$

$$g_{q,p}(k,l) = (k^2 + l^2)^{1-\sigma} h_{q,p}^*(k,l) = (k^2 + l^2)^{-\frac{\sigma}{2}} \exp\left(-i\left[\frac{p}{2} \ln(k^2 + l^2) - q \arctan\left(\frac{l}{k}\right)\right]\right) \quad (91)$$

Nous verrons, en 2.3.6, une illustration des résultats que l'application de cette transformation inverse permet d'obtenir.

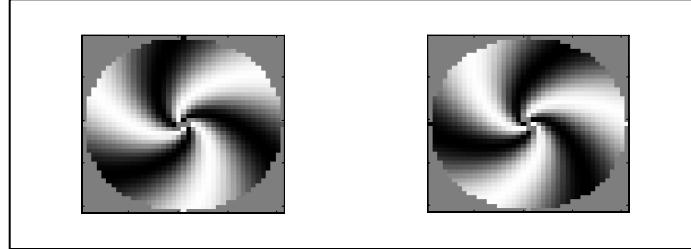


Figure 2.14 : Les parties réelles (à gauche) et imaginaires (à droite) de  $h_{p,q}$  avec  $p=2$ ,  $q=3$ , et  $\sigma_0=2$ .

### 2.3.5. Estimation des paramètres de mouvement

Une autre propriété théorique très intéressante de la transformée de Fourier Mellin provient de la possibilité d'estimer l'orientation et la taille des objets analysés par rapport à des objets de référence. En effet, connaissant la TFM d'une forme de référence, et celle d'une forme identique à une similitude près, il est possible de retrouver les paramètres de la similitude, c'est-à-dire le couple  $(\alpha, \beta)$ .  $\alpha$  représente le facteur d'échelle et  $\beta$  représente l'angle entre les deux formes. [Adam 2001a]. Soient deux fonctions (les formes)  $f$  et  $g$  identiques aux facteurs de rotation et de changement d'échelle près, le théorème du retard s'applique et la théorie nous donne alors :

$$M_g(q,s) = \alpha^{-s} \exp(iq\beta) M_f(q,s) \quad \forall q \in \square \text{ et } s = \sigma + i\nu \in \square \quad (92)$$

Donc :

$$M_g(q,s) - \alpha^{-s} \exp(iq\beta) M_f(q,s) = 0 \quad \forall q \in \square \text{ et } s = \sigma + i\nu \in \square \quad (93)$$

En sommant pour tout  $q$  et  $s$  appartenant à leur domaine de définition, alors :

$$\int_s \sum_q \left| M_g(q,s) - \alpha^{-s} e^{iq\beta} M_f(q,s) \right|^2 ds = 0 \quad (94)$$

Dans cette expression, si les valeurs estimées de  $\{M_f(q,s)\}$  et de  $\{M_g(q,s)\}$  sont disponibles, l'estimation des paramètres de mouvement entre les deux fonctions  $f$  et  $g$ , identiques au facteur d'échelle et de rotation près, consistera alors à minimiser le critère quadratique :

$$E_{f,g}(\alpha, \beta) = \int_s \sum_q \left| \tilde{M}_g(q,s) - \alpha^{-\sigma + i\nu} e^{iq\beta} \tilde{M}_f(q,s) \right|^2 ds \quad (95)$$

L'estimation des paramètres de mouvement consiste donc à calculer :

$$(\alpha^*, \beta^*) = \text{Arg min}_{(\alpha, \beta)} E_{fg}; \alpha \in \mathbf{R}_+, \beta \in [0, 2\pi[ \quad (96)$$

En terme d'application en reconnaissance de formes, il faudra donc disposer, pour chacune des formes possibles (les différentes fonctions  $f$ ), des valeurs des  $\{M_f(q,s)\}$ . Lorsqu'une forme  $g$ , identique aux paramètres de similitude près à  $f$  sera détectée, il sera possible, par calcul des  $\{M_g(q,s)\}$ , d'en identifier l'échelle et l'orientation par minimisation de (95). Nous reviendrons sur l'algorithmique d'application de cette technique dans la partie 2.3.6.3.

### 2.3.6. Algorithmes de mise en œuvre

La section précédente a mis en évidence le fait que les propriétés de la TFM présentent des caractéristiques intéressantes pour la représentation d'une forme indépendamment de sa position, de sa taille et de son orientation. Afin d'illustrer l'implémentation de cette TFM dans le cadre de notre problème d'analyse de document, nous proposons dans cette partie l'ensemble des algorithmes développés pour la mise en œuvre de la théorie. Nous développons dans ce cadre de manière séparée deux modes de fonctionnement de notre outil de détection/reconnaissance de formes. Dans une première partie, nous abordons l'implémentation de la TFM pour la reconnaissance des formes isolées, parfaitement segmentées. Puis, dans un second temps, nous développons l'implémentation du « mode filtrage » qui vise la détection et la reconnaissance de formes connectées entre-elles ou connectées à d'autres parties graphiques du document. Enfin, dans un troisième paragraphe, nous présentons l'algorithme d'estimation des paramètres de similitudes des formes.

#### 2.3.6.1. Algorithme d'extraction des IFM sur des formes isolées

Dans ce paragraphe, nous exposons l'algorithme développé afin d'extraire les invariants de Fourier Mellin d'une forme préalablement isolée, en exploitant les équations (83), (86) et (87). Si théoriquement l'extraction de ces invariants peut se faire à partir d'une image en niveaux de gris, nous ne l'avons appliquée dans le cadre de l'application France Telecom qu'à des images binaires pour les raisons que nous avons évoquées dans l'étude de la bibliographie. L'algorithme suivant est donc appliqué à une image préalablement binarisée, en utilisant le principe décrit dans [Adam 2000a].

Les entrées de l'algorithme sont les suivantes :

- L'imagette binaire d'une forme connexe isolée. Cette imagette n'a aucune contrainte de taille. Néanmoins, seuls seront pris en compte les pixels d'une fenêtre de dimension  $TailleMasque * TailleMasque$
- La valeur des pixels correspondant à la forme et au fond
- Les intervalles de variations des paramètres de la TFM ( $P$  et  $Q$ ), permettant la définition de l'ensemble A donné en (89).

Le choix de  $P$  et de  $Q$  engendrera le calcul de  $2PQ+P+Q+1$  filtres, chacun d'eux générant un descripteur complexe. De ce fait, compte tenu de l'absence d'intérêt des valeurs  $I_f(0, \sigma)$  qui vaut 1 et de la partie imaginaire de  $I_f(1, \sigma)$  qui vaut 0, le modèle extrait sera donc constitué de  $4PQ+2P+2Q-1$  valeurs réelles comprises entre  $-1.0$  et  $1.0$ . Cela impliquera, en terme de complexité, un nombre d'opérations élémentaires (multiplication de complexes) de  $(2PQ+P+Q+1)r^2$ , où  $r$  vaut  $TailleMasque$ .

**DEBUT**

Calcul des filtres  $h_{p,q}$  de Fourier Mellin (équation (87)) avec les  $P$  et  $Q$  choisis.

Extraction de la masse connexe

Calcul du centre de gravité de la forme – nous utilisons pour ce faire une technique basée sur les histogrammes de projection (cf. 2.2.3.3) –

Recopie de la forme dans une imagerie de taille TailleMasque, de manière à centrer  $C_g$  sur le centre de l'image

**Pour** Chacun des filtres **faire**

**Début**

SOMME=0

**Pour** chacun des pixels de l'image **faire**

**Début** (*Calcul des descripteurs*)

**Si** le pixel est de la forme **alors**

**Début**

Ajouter le coefficient correspondant du filtre à SOMME

**Fin**

Stocker SOMME dans la liste de descripteurs correspondant à la forme

**Fin**

**Fin**

Recherche des valeurs  $M_f(0,\sigma)$  et  $M_f(1,\sigma)$

**Pour** Chacun des filtres **faire**

**Début**

Calculer les parties réelles et imaginaires de  $I_f(p, q)$  par (83)

**Fin**

**FIN**

*Algorithme 1 : Algorithme d'extraction des invariants de Fourier Mellin sur une forme binaire isolée*

L'ensemble des résultats concernant les formes isolées qui seront présentés dans le chapitre suivant sera issu de l'application de cet algorithme.

### 2.3.6.2. Algorithme d'extraction des IFM sur des images quelconques

Dans l'Algorithme 1, la TFM est appliquée au centre de gravité de la forme (en tant que CdD). Cette technique est parfaitement adaptée à la modélisation de formes bien conditionnées (c'est-à-dire déconnectées de toute autre forme), puisque le calcul des coordonnées du centre de gravité est alors facile. Ce n'est pas le cas pour des formes connectées puisque le centre de gravité de deux formes collées n'a plus de sens. Dans un tel cas de figure, la modélisation de la forme passe alors par un calcul de modèle de Fourier Mellin en chacun des pixels de l'imagerie, qui seront tous considérés comme des centres de gravité potentiels. Ce principe se rapprochant d'un filtrage des pixels candidats, nous appellerons ce principe « **mode filtrage** » dans la suite de ce mémoire. En fait, la bonne faculté de détection de cet opérateur est principalement due à la décroissance du module des filtres  $h_{p,q}(\dots)$  lorsque la distance au centre de gravité augmente : dans la plupart des configurations où les caractères et symboles sont connectés les uns aux autres, ce problème se produit uniquement au voisinage de l'enveloppe externe de la forme. Dans de tels cas, la contribution d'une autre forme connectée sur le calcul appliqué au centre de gravité de la forme courante est minimale, et n'affecte donc pas trop le processus de la classification. Notons que l'approche correspondant à l'algorithme suivant n'est pas uniquement applicable au traitement de formes collées, mais aussi au traitement de texture pouvant

apparaître dans différentes orientations. Les résultats présentés dans [Ravichandran 1995] illustrent une telle application, avec par exemple l'impressionnante reconnaissance de textures correspondant à la peau d'un zèbre.

Pour appliquer la TFM sans utiliser de CdD particulier, il est nécessaire d'avoir recours à un produit de convolution entre l'image et l'ensemble des filtres de Fourier Mellin (équations (86) et (87)). Ce procédé entraîne un coût de calcul rédhibitoire. Or, il est bien connu qu'un produit de convolution dans le domaine spatial équivaut strictement à un produit dans le domaine spectral. Par un calcul de transformée de Fourier rapide (connu sous le nom de **FFT** : **F**ast **F**ourier **T**ransform), il est donc possible de s'affranchir de ce problème de complexité en transformant le calcul du produit de convolution par un simple produit, en ayant au préalable appliqué la FFT à l'image. Néanmoins, les filtres  $h_{p,q}(\dots)$  et l'image n'ayant pas les mêmes tailles, il est donc nécessaire de « réarranger » les données. De plus la FFT implique des contraintes concernant la taille des images qui doit être une puissance de 2. Les transformations appliquées aux images et aux filtres, qui ne sont pas d'ordre théorique mais du domaine de l'implémentation, sont présentées après l'algorithme, en Figure 2.15 et Figure 2.16.

Les paramètres de cet algorithme sont l'imagerie et les intervalles de variations des paramètres de la TFM (les valeurs de  $P$  et  $Q$ )

**DEBUT**

Calcul des filtres  $h_{p,q}$  de Fourier Mellin (équation (87)) avec les  $P$  et  $Q$  choisis.

Réarrangement de l'image suivant la Figure 2.16

Application de la FFT à l'image

**Pour** Chacun des filtres **faire**

**Début**

    Réarrangement du filtre suivant la Figure 2.15

    Application de la FFT au filtre réarrangé

    Produit complexe de la TF de l'image et de la TF du filtre

**Fin**

Récupération des  $M_i(p,q)$  dans le tableau résultant des produits

Calcul des invariants  $I_i(p,q)$  suivant le même procédé que dans Algorithme 1

Réarrangement de l'image de façon à revenir à sa taille originale

Retourner un tableau tridimensionnel contenant  $2PQ+P+Q-2$  invariants par pixel de l'image

**FIN**

*Algorithme 2 : Algorithme d'extraction des invariants de Fourier Mellin sur une imagerie quelconque*

Concernant le réarrangement des filtres et de l'image pour l'utilisation de la FFT, l'approche suivante est utilisée. Soient  $L$  et  $H$  la largeur et la hauteur de l'image, et  $T$  la taille des filtres, valant  $2r_{max}+1$  (86). Notons alors  $Lm$  (respectivement  $Hm$ ) la première puissance de 2 supérieure à  $L$  (respectivement  $H$ ), les figures suivantes montrent les étapes du réarrangement des filtres et de l'image afin de pouvoir utiliser la FFT.



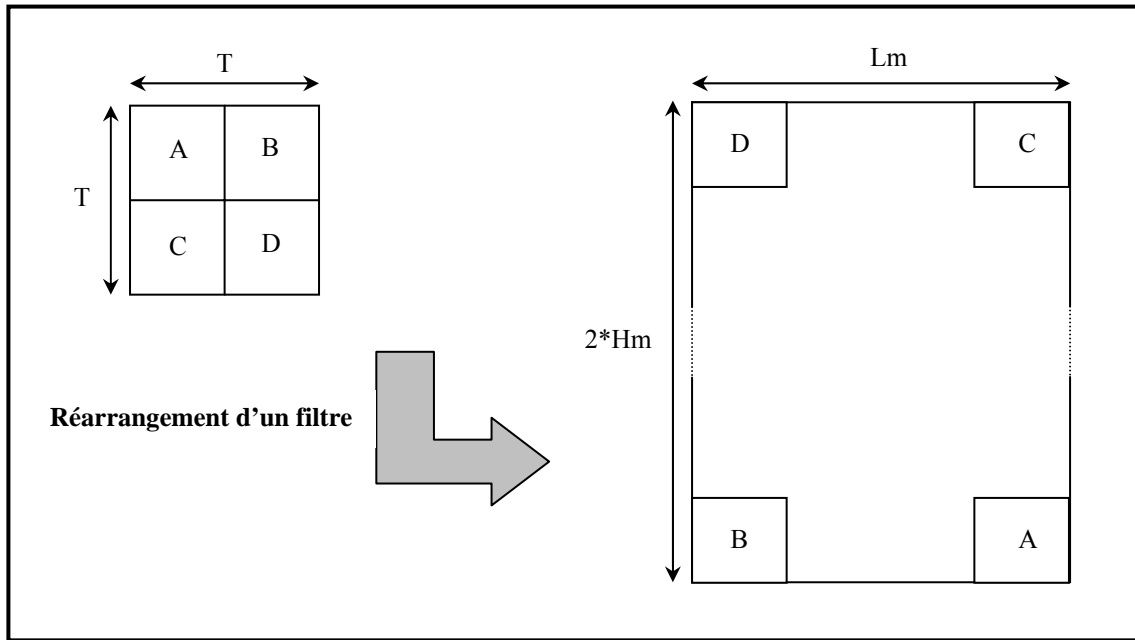


Figure 2.15 : Schéma de principe du réarrangement des filtres de FM

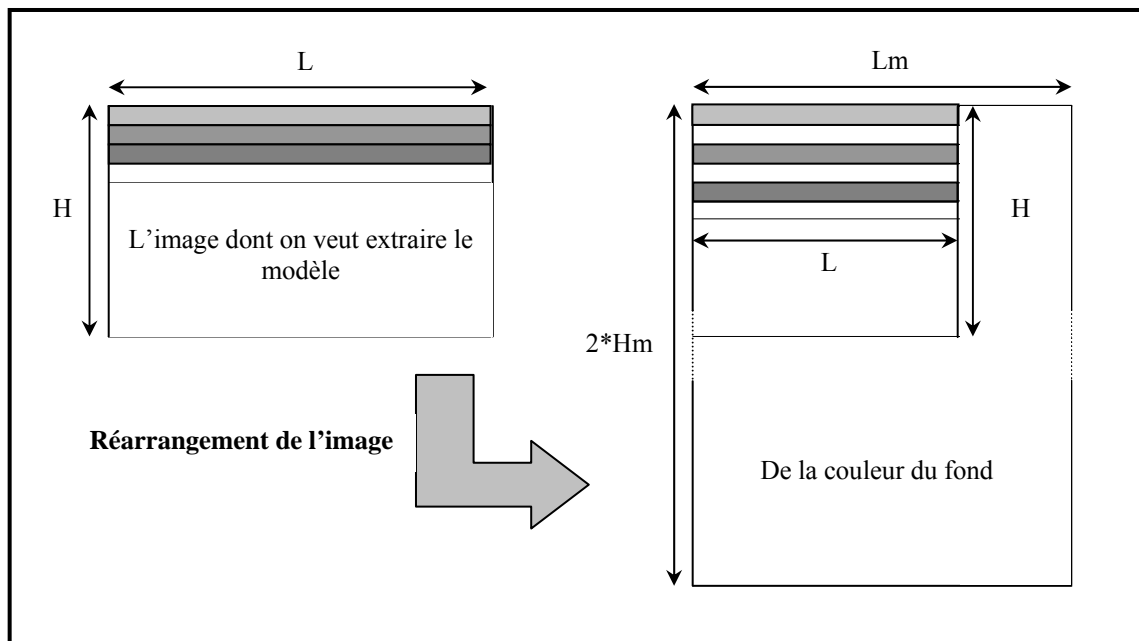


Figure 2.16 : Schéma de principe du réarrangement de l'image

Les résultats que nous présentons par la suite en ce qui concerne les formes connectées sont issus de l'application de cette algorithmique.

### 2.3.6.3. Algorithme de calcul des paramètres de mouvement

En 2.3.5, nous avons montré qu'il était possible d'estimer l'orientation d'une forme (voir équation (96)), en disposant des  $M_g(p,q)$  issus de celle-ci ainsi que des  $M_f(p,q)$  de la même forme dans une position de référence (horizontale par exemple). Néanmoins, nous avons vu qu'une telle estimation passait par l'optimisation d'un critère quadratique. Malheureusement, il s'avère que la mise en œuvre

numérique de la TFM par application de (87) rend ce critère non convexe : il existe plusieurs *minima* locaux, comme en témoigne la Figure 2.17 qui correspond à l'erreur quadratique entre forme de référence (lettre « A » à gauche « E » à droite) et une forme de même taille et tournée de 100 degrés vers la gauche.

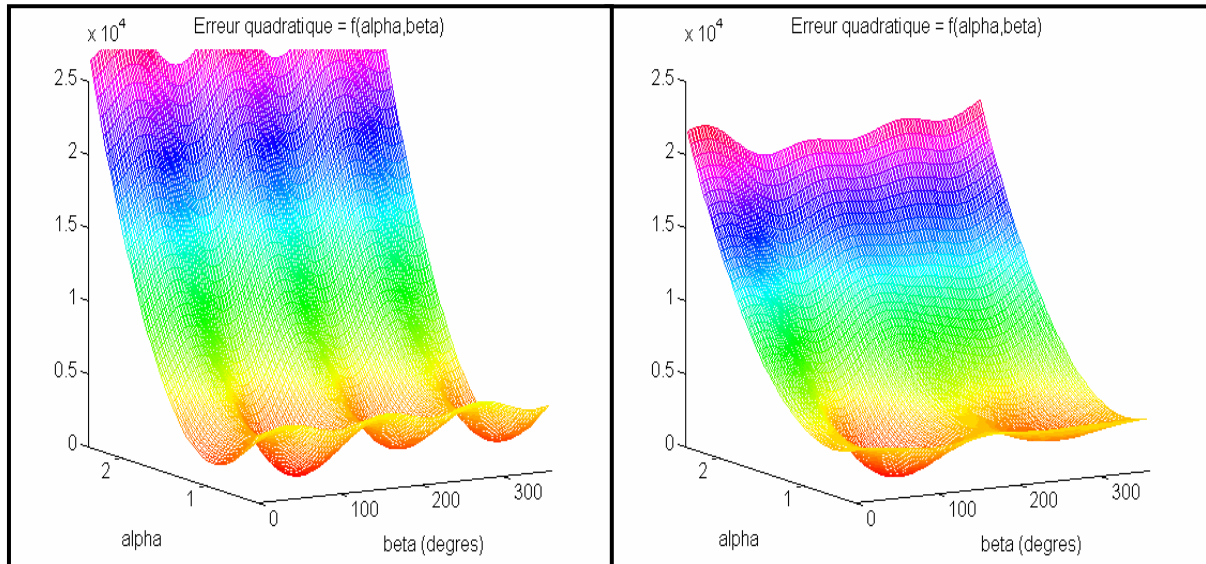


Figure 2.17 : Deux illustrations de la non convexité du critère  $E_{f,g}(\alpha\beta)$  défini en (95)

Du fait de cette évidente non convexité, il est nécessaire d'adopter une méthode d'optimisation s'affranchissant de ce problème. Cette méthode ne doit pas être basée sur un calcul de gradient. Quatre types d'approches pouvaient donc être utilisés :

- Les méthodes énumératives, qui consistent à examiner les valeurs prises par la fonction en chacun des points de l'espace un par un. Bien que la simplicité de telles procédures soit attrayante, celles-ci doivent ici, comme dans la majeure partie des problèmes d'optimisation, être abandonnées à cause de la taille de l'espace des possibilités.
- Les méthodes basées sur des combinaisons de plusieurs solutions, qui optimisent par des essais-erreurs en calculant des combinaisons linéaires des coordonnées des différents points afin de faire évoluer ces points vers la bonne solution. C'est par exemple le cas de la méthode du *simplex* qui se révèle souvent efficace, mais aussi très lente.
- Les méthodes aléatoires, qui explorent aléatoirement l'espace des solutions et mémorisent le meilleur élément. De telles approches sont peu robustes et manquent d'efficacité dans la plupart des cas.
- Les méthodes dites pseudo-aléatoires, qui utilisent le hasard, au même titre que les méthodes aléatoires, mais sous contrôle du but à atteindre et non de manière totalement désordonnée. Les deux méthodes de cette catégorie sont le recuit simulé et les algorithmes génétiques.

Parmi ces choix, nous avons opté pour deux techniques issues de deux catégories d'approches différentes : la méthode du *simplex*, issue des travaux de Nelder et Mead [Nelder 1965] pour la robustesse que la littérature lui accorde, et les algorithmes génétiques [Goldberg 1989] qui sont plus rapides mais qui n'assurent pas l'obtention de la bonne solution. Une description de ces approches est donnée en Annexe A. Notons que dans le cas de la méthode du *simplex*, nous avons utilisé les outils des *Numerical Recipes*, alors que dans le cas des algorithmes génétiques, nous avons utilisé une bibliothèque générique développée par notre équipe [Adam 1999a]. Quelle que soit la méthode d'optimisation utilisée, l'algorithme d'estimation des paramètres de mouvement est le même. Ses paramètres sont les suivants :

- Les intervalles de variations des paramètres de la TFM (les valeurs de  $P$  et  $Q$ )
- L'étiquette de la forme dont les paramètres de mouvement sont estimés.
- Une base contenant les descripteurs de Fourier Mellin (les  $M_f(p,q)$ ) issus de la même forme dans une position de référence.
- Les descripteurs de Fourier Mellin de la forme dont les paramètres de mouvement vont être estimés.

A partir de ces paramètres, l'algorithme d'estimation des paramètres de mouvement consiste simplement à appeler l'une des deux méthodes d'optimisation, en évaluant les fonctions par (95). Notons toutefois que pour chacune des méthodes d'optimisation, plusieurs paramètres sont à régler ; nous n'avons pas poussé les investigations dans ce domaine.

### 2.3.7. Résultats qualitatifs visuels

Dans cette partie, après avoir présenté la théorie, puis les différents algorithmes mettant en œuvre cette théorie, nous proposons une illustration des propriétés de la TFM sous forme de résultats qualitatifs, avant d'aborder dans le chapitre suivant des résultats quantitatifs.

#### 2.3.7.1. Illustration de la stabilité

Cette première section vise à mettre en évidence la stabilité des invariants de Fourier Mellin. Pour ce faire, un exemple de description invariante pour 5 caractères différents est donné en Figure 2.19. Cette description correspond aux 13 lettres multi-orientées (de A à E) présentées en Figure 2.18. Ces formes ont été analysées par le banc de filtres  $\{h_{q,p}(.,.)\}$ , avec  $\sigma = 1$ ,  $r_{\max} = 20$ ,  $Q = 3$  et  $P = 2$ . Compte tenu des remarques précédentes, ces valeurs de  $Q$  et  $P$  conduisent à l'utilisation de 18 filtres, et engendrent alors un vecteur de 33 paramètres invariants réels et non redondants. Les mêmes formes ont aussi été analysées à un facteur d'échelle de 0,5, simplement obtenues par sous-échantillonnage de l'image originale de la Figure 2.18, afin de mettre en évidence la stabilité des invariants vis-à-vis du changement d'échelle. Certes, cette stratégie de sous-échantillonnage n'apparaît pas comme

parfaitement représentative de la multi-résolution réellement rencontrée dans le cadre de l'analyse documentaire. Néanmoins, nos expérimentations mettent en évidence le fait que cette démarche d'évaluation met le dispositif dans des conditions plus difficiles que les conditions réelles puisque le sous-échantillonnage induit une perturbation des formes supérieure à celle que nous rencontrons dans la réalité, lors du traitement de formes de tailles variables. Pour cette analyse de formes multi-échelles, nous avons donc délibérément opté pour la démarche du « qui peut le plus peut le moins ». Toutefois cette démarche, qui vise à démontrer la faisabilité du procédé, n'est pas la seule stratégie d'étude de la robustesse des invariants de Fourier Mellin au changement d'échelle. La réalité industrielle et les bases issues de documents techniques viendront étayer ce discours autour de la stabilité des invariants.

La Figure 2.19 montre les primitives invariantes obtenues, chaque ligne correspondant aux invariants pour une forme donnée. Par exemple, la première ligne représente la superposition des 33 invariants pour chacune des 26 formes "A" disponibles en 13 orientations et 2 tailles différentes. Notons que sur cette figure, l'ordonnancement des invariants n'a aucun sens particulier, et correspond à un balayage quelconque des valeurs de  $q$  et  $p$ . Toutefois, signalons que le pic en position 18 correspond au coefficient (réel)  $I_f(q = 0, s = \sigma)$ .

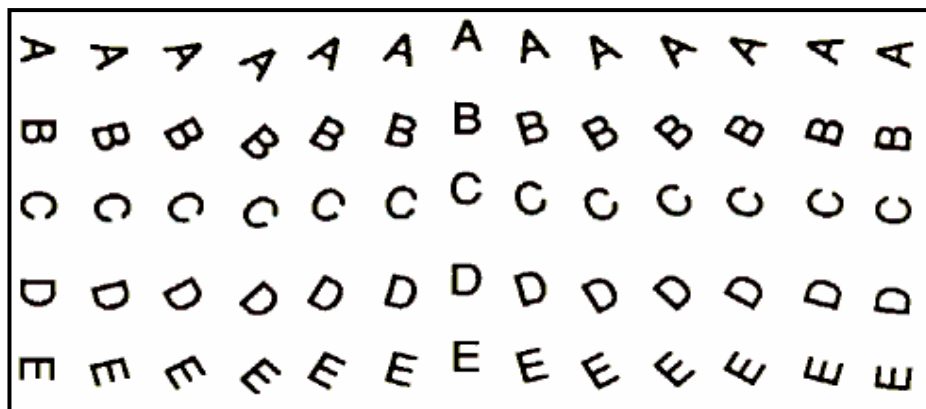


Figure 2.18 : Un extrait d'une base synthétique de caractères multi-orientés

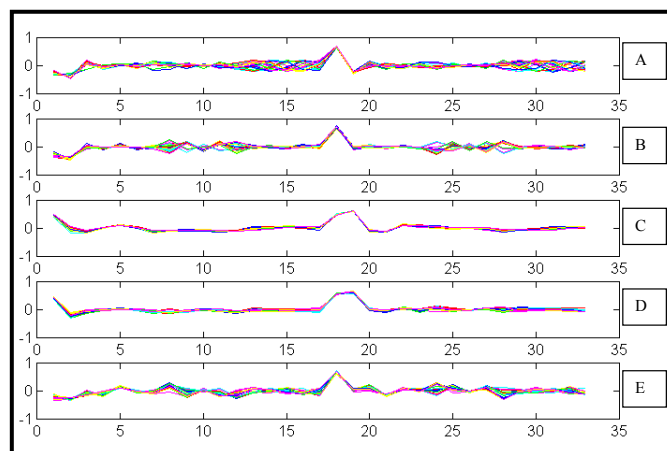


Figure 2.19 : Superposition des invariants issus des différentes orientations et des différentes tailles

### 2.3.7.2. Illustration de la complétude

Nous avons évoqué précédemment la propriété de complétude de la TFM. Sur la Figure 2.20, nous illustrons cette propriété, en montrant la reconstruction par TFM discrète inverse ((90) et (91)) de quelques formes (celles de la Figure 2.18) à partir d'une analyse basée sur un ensemble de 61 filtres ( $Q = 5$  et  $P = 5$ ), donnant 119 invariants non redondants par forme.

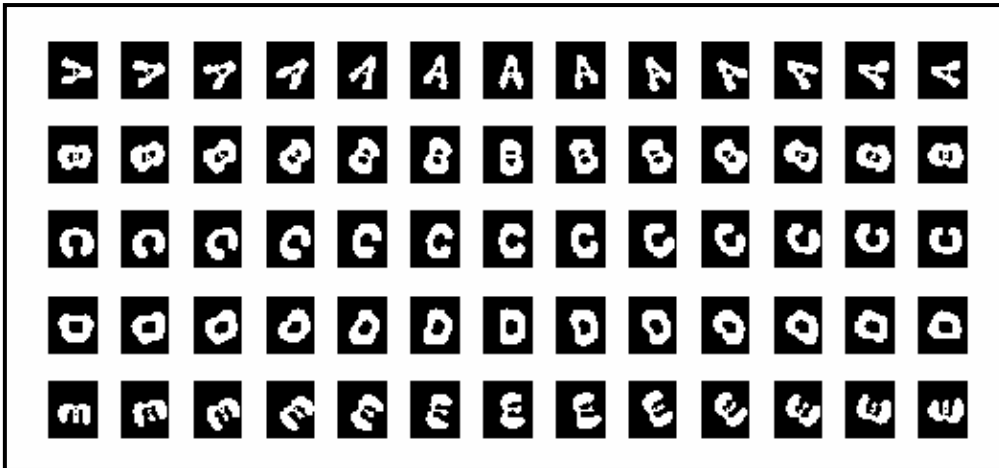


Figure 2.20 : Illustration de la complétude de la TFM : reconstruction par la TFM discrète inverse

Notons toutefois qu'ici, pour des questions de lisibilité, nous montrons l'effet de la reconstruction sous la forme d'une image binaire. Ce n'est évidemment pas le résultat brut de l'application des équations de la TFM inverse, qui conduit à l'obtention de valeurs réelles. Une binarisation dont le seuil a été évalué manuellement a été appliquée à l'image résultat. Rappelons que plus l'ordre utilisé est important, plus la reconstruction est précise.

### 2.3.7.3. Illustration de l'estimation de l'orientation

Nous avons montré précédemment qu'il était possible d'estimer les paramètres de mouvement d'une forme, à condition de disposer de descripteurs issus de la même forme dans une position de référence. Sur la Figure 2.21, nous montrons les résultats d'une estimation en utilisant les principes de 2.3.6.3. Les formes de la Figure 2.18 sont ici analysées, en utilisant comme formes de référence les formes de la colonne de gauche. Sur la Figure 2.21, des groupes d'étoiles sont facilement identifiables sur le cercle trigonométrique. Chacun de ces groupes correspond à l'estimation commune des 5 formes différentes. 12 groupes se distinguent nettement. Deux groupes (proches de 0) sont trop proches pour pouvoir les distinguer graphiquement. Néanmoins, les valeurs numériques correspondantes sont correctes. Les quelques erreurs qui peuvent être constatées sur la partie basse du cercle correspondent à des erreurs de reconnaissance. Cela est dû au fait que nous n'avons utilisé qu'un prototype par forme pour l'apprentissage lors de ces tests.

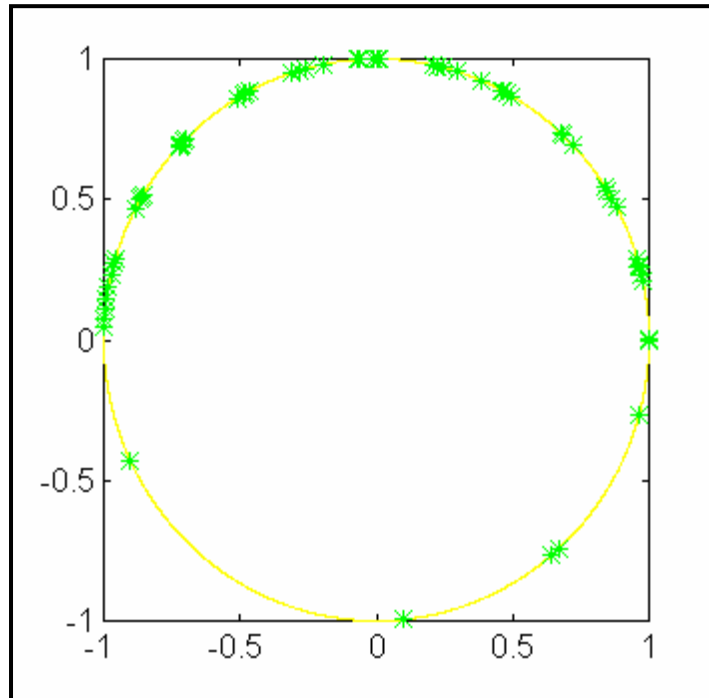


Figure 2.21 : Illustration de l'estimation de l'orientation sur le cercle trigonométrique

#### 2.3.7.4. Illustration du traitement des formes connectées

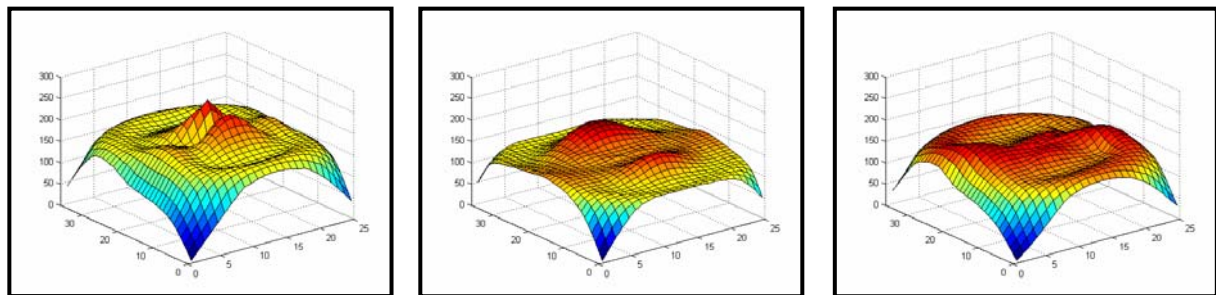
Dans cette partie, nous montrons une illustration de l'application de l'Algorithme 2, qui consiste à extraire un modèle de Fourier Mellin à partir d'une image, sans baser le calcul sur l'information composante connexe. Ce mode filtrage consiste, comme nous l'avons expliqué en 2.3.7.1, à assimiler chaque pixel d'une image au centre de gravité potentiel d'une forme. Il vise à reconnaître les formes qui sont connectées au réseau ou entre elles sur les documents techniques. Sur la Figure 2.22, nous montrons une image d'un caractère « 8 » que nous avons traitée en utilisant un tel principe. En appliquant l'Algorithme 2 à cette image, nous obtenons un modèle numérique où chacun des pixels de l'image est représenté par 33 invariants de Fourier Mellin ( $\sigma = 1$ ,  $r_{\max} = 20$ ,  $Q = 3$  et  $P = 2$ ). Cela nous a permis de comparer les jeux d'invariants extraits en chacun des pixels par le mode filtrage à ceux extraits avec la méthode classique (2.3.7.1). Sur la Figure 2.23, nous montrons les résultats obtenus. Nous y présentons des distances euclidiennes (normalisées entre 0 et 255) calculées entre les invariants issus de chacun des pixels de l'image du « 8 » et des invariants calculés par la méthode classique sur l'image d'un chiffre « 8 » (Figure 2.23a), d'un chiffre « 0 » (Figure 2.23b), et d'un chiffre « 1 » (Figure 2.23c).

Sur la Figure 2.23a, un pic correspondant au centre de gravité du « 8 » peut facilement être remarqué. Il correspond à une distance nulle (255 sur l'image) puisque la même image est utilisée pour le filtrage et pour l'extraction classique. La présence du deuxième pic, dont l'amplitude est moins grande, s'explique par la symétrie imparfaite du « 8 » par rapport à son centre de masse et par l'invariance à la rotation du modèle. Sur la Figure 2.23b, le filtrage permet de distinguer la présence des deux boucles

du « 8 », en assimilant approximativement celles-ci à des « 0 », ce qui prouve cette fois l'invariance du modèle au changement d'échelle. Enfin, sur la Figure 2.23c le filtrage assimile certains des points du contour du « 8 » à la présence de « 1 », essentiellement au niveau des jonctions entre les deux boucles, où il y a des parties droites assimilables à des « 1 » alors que les pixels des occlusions du « 8 » semblent loin de « 1 ».



Figure 2.22 : Une image de caractères pour l'illustration du mode filtrage



a : Recherche d'un « 8 »

b : Recherche d'un « 0 »

c : Recherche d'un « 1 »

Figure 2.23 : Illustration des résultats obtenus par le « mode filtrage »

Ces résultats prouvent l'efficacité de l'Algorithme 2. Nous illustrerons son exploitation dans le cadre de la reconnaissance de formes connectées sur des documents techniques en 2.4.3.2.

### 2.3.8. Conclusion

Dans ce chapitre, nous avons présenté un nouveau modèle de représentation de formes basé sur la transformée de Fourier Mellin (TFM). Après un rapide rappel des notions de base nécessaires à la construction de cette transformée, nous l'avons présentée d'un point de vue théorique, en montrant aussi comment celle-ci est appliquée à la structure bidimensionnelle discrète des images. Les propriétés intéressantes dans le cadre de notre étude de cette TFM ont ensuite été exposées du point de vue mathématique. Puis, nous avons montré de façon plus intuitive, grâce à quelques figures, les principaux avantages de ce modèle, à savoir :

- La stabilité des invariants vis à vis des transformations que sont la translation, la rotation et le changement d'échelle des formes

- La possibilité d'extraire les paramètres de mouvement des formes en exploitant la complétude de la TFM, ce qui se montrera extrêmement intéressant quant à la reconstruction de chaînes de caractères
- La possibilité de traiter des images en mode « filtrage », ce qui permettra d'extraire d'un « amas » de formes les modèles correspondant approximativement à chacune d'elles.

Dans le chapitre qui suit, nous allons maintenant présenter des résultats quantitatifs obtenus par l'intermédiaire de ce modèle, en les comparant à des modèles reconnus de la littérature.



## 2.4. Evaluation quantitative des modèles : les résultats obtenus

### 2.4.1. Introduction

Pour avoir un point de vue objectif sur l'efficacité d'un modèle de représentation de forme, nous avons déjà évoqué le fait qu'il est nécessaire de le comparer à d'autres modèles, sur les mêmes bases et en suivant le même protocole. Dans ce chapitre, nous présentons l'évaluation de notre modèle de Fourier Mellin, en le comparant aux moments invariants de Zernike de Belkasim (2.2.3.5), et aux sondes circulaires de Lefrère (2.2.3.7). Le choix de ces deux approches repose d'une part, sur la supériorité reconnue des moments de Zernike dans la littérature, et sur la complémentarité entre l'aspect global de ces moments et l'aspect plus local apporté par une approche comme celle des sondes circulaires.

Cette évaluation est décomposée en trois parties distinctes, articulées comme la présentation des résultats qualitatifs. Nous verrons d'abord les résultats obtenus en ce qui concerne la reconnaissance de formes isolées, sur des formes de synthèse, sur des formes issues des documents France Telecom, sur des formes issues de documents EDF et sur des formes de la base NIST<sup>8</sup>. Puis, les résultats obtenus par le mode « filtrage », là encore sur des données de synthèse et sur quelques données issues de plans France Telecom. Enfin, nous montrerons quelques résultats concernant l'estimation de l'orientation, résultats moins étayés à cause de la difficulté de constitution de bases.

### 2.4.2. Les résultats obtenus pour les formes isolées

Dans cette partie, nous montrons les résultats que permet d'obtenir le modèle de Fourier Mellin pour reconnaître des formes isolées. Pour effectuer cette reconnaissance, l'extraction du modèle n'est pas suffisante. Comme le montre la Figure 2.3, il est nécessaire d'y associer une méthode d'appariement entre l'instance du modèle issu d'une forme inconnue et les modèles contenus dans une base d'apprentissage. Dans le cas d'un modèle numérique, cet appariement est appelé classification et il est réalisé par l'intermédiaire d'un classifieur. C'est pourquoi dans un premier temps, nous présenterons rapidement les deux classifieurs que nous avons utilisés, à savoir le *Learning Vector Quantization* (LVQ) et les *K-Plus-Proches-Voisins* (KPPV), en motivant ces choix. Puis, nous présenterons chacune des bases utilisées pour cette évaluation et nous rappellerons les modèles évalués avant de donner les tableaux de résultats critiqués.

#### 2.4.2.1. Les classifieurs pour l'évaluation

La stratégie de la classification utilisée dans le cadre de notre évaluation est basée sur le test de deux classifieurs différents : le *Learning Vector Quantization* (LVQ) et les *K-Plus-Proches-Voisins*

---

<sup>8</sup> National Institute of Standards and Technology

(KPPV). Le choix de ces classifieurs a été motivé par deux arguments : d'abord, la technique des KPPV représente une bonne référence sur laquelle une évaluation peut être effectuée objectivement ; ensuite, l'approche du LVQ est une technique intéressante du point de vue de la mise en œuvre. Dans notre contexte industriel les réseaux de neurones présentent en effet la particularité intéressante d'être rapides en décision. Nous rappelons ci-dessous rapidement le fonctionnement de tels classifieurs.

*i. Le Learning Vector Quantization*

L'algorithme **LVQ** (*Learning Vector Quantization*)<sup>9</sup> est issu de travaux de T. Kohonen [Kohonen 1989], chercheur Finlandais connu pour ses travaux sur les mémoires associatives. Le LVQ est un classifieur supervisé qui consiste donc à déterminer la classe d'un élément inconnu en exploitant une base d'apprentissage contenant des représentants préalablement étiquetés de chacune des classes. Le LVQ fait partie des approches connexionnistes, c'est un réseau de neurones à trois couches. La première couche représente les vecteurs d'entrées, c'est-à-dire le modèle de la forme ; il possède classiquement autant de cellules neuronales que de valeurs dans le vecteur modèle. La deuxième couche est totalement connectée à la première et chaque unité de cette couche est un neurone linéaire ; sa sortie est calculée sur la base d'une combinaison linéaire de ses entrées. Les neurones de la seconde couche sont pré-affectés aux différentes classes, ils peuvent être représentés comme un vecteur de  $R^N$  (le même espace que celui des exemples). Chaque classe possède de ce fait un *pool* de neurones bien identifiés dans cette couche, *pool* qui a plus ou moins d'activité lors de la phase de décision pour une entrée inconnue. Le *pool* ayant le plus d'activité est alors relié à la couche de sorties, qui contient autant de cellules que de classes. La décision est donc prise par simple comparaison des activités des cellules cachées. Pour construire un tel réseau, il est évidemment nécessaire d'avoir recours à une phase d'apprentissage supervisé. Cet apprentissage joue uniquement sur les poids reliant la couche d'entrée et la couche cachée. Divers algorithmes d'apprentissage ont été proposés [Odorico 1997], néanmoins, le principe de base est resté le même. Aussi ne présentons-nous ici que la règle d'apprentissage du LVQ1.

Notons :

- $W_i$  : la  $i^{\text{ème}}$  coordonnée du représentant neuronal à mettre à jour
- $\delta(c)$  : +1 si l'élément courant est de même classe que le représentant  
-1 si l'élément courant est d'une autre classe que le représentant
- $\eta(t)$  : le coefficient d'apprentissage, décroissant dans le temps
- $X_i$  : la  $i^{\text{ème}}$  coordonnée de l'exemple d'apprentissage.

Alors, la règle de modification des poids durant l'apprentissage sera :

---

<sup>9</sup> quantification vectorielle supervisée

$$\Delta W_i = \delta(c).\eta(t).(X_i - W_i) \quad (97)$$

Dans cette équation (97), le principe est que chaque représentant se rapproche de sa classe et s'éloigne des autres. Le principe de base du LVQ peut donc être assimilé à une version de 1PPV hiérarchique. De ce fait ses performances sont bonnes, essentiellement en généralisation, et il est plus rapide en décision que le classifieur basé sur la technique des k Plus Proches Voisins. D'un point de vue logiciel, nous avons utilisé une version sur étagère<sup>10</sup> du LVQ. Dans cette version, l'algorithme d'apprentissage est le LVQ3, qui est une version légèrement modifiée de la technique exposée ci-dessus, mais qui garde exactement la même philosophie.

## ii. *Les k-Plus-Proches-Voisins*

Le classifieur des k-Plus-Proches-Voisins (kPPV) est basé sur une idée naturelle : si une forme inconnue a un modèle de représentation proche de celui d'une autre, les deux formes sont probablement de la même classe. L'algorithme de base d'un kPPV consiste donc à calculer la distance entre le nouvel objet et tous ceux contenus dans la base d'apprentissage. Les k plus proches voisins du prototype inconnu sont ensuite examinés afin de prendre la décision. Pour ce faire, plusieurs stratégies sont possibles, en sélectionnant par exemple la classe la plus représentée parmi les k votes. Les paramètres d'un tel algorithme sont au nombre de trois :

- La distance utilisée : nous avons choisi d'utiliser la distance euclidienne, plus efficace que celle de Hamming sur la base des quelques tests que nous avons menés. De plus amples investigations concernant ce choix, et concernant par ailleurs l'ensemble du problème de la classification, constituent l'une des priorités de nos travaux à venir. L'exploitation d'un classifieur neuronal particulier, nommé « réseau d'yprels » [Ribert 1999] et développé dans notre laboratoire semble à ce titre une piste très intéressante. Néanmoins, rappelons que l'objectif de notre travail est essentiellement ici la validation de la représentation des formes dans l'espace de Fourier Mellin.
- La méthode de décision : si le vote majoritaire est probablement la technique la plus utilisée dans la littérature, de très nombreuses variantes existent. Il est par exemple possible d'envisager une pondération de chaque voisin en fonction de son rang ou de la distance le séparant de l'objet inconnu.
- Le nombre k de voisins à prendre en compte : son choix revêt une grande importance. D'une manière générale il doit être pris significativement plus petit que le nombre d'éléments de la base d'apprentissage, afin d'éviter de se prononcer trop souvent en faveur de la classe majoritaire dans celle-ci. Nous avons, dans la plupart de nos tests, utilisé  $k=1$ , en étant conscient qu'un tel choix ne permettrait pas l'obtention du meilleur compromis taux de reconnaissance / taux de confusion. Le but d'un tel choix était plutôt d'obtenir une référence.

Notons enfin que nous avons utilisé une version « brute » de ce classifieur, sans optimisation, malgré les nombreuses propositions disponibles dans la littérature (les kppv hiérarchiques par exemple). Le lecteur intéressé trouvera des pointeurs relatifs à de tels travaux dans [Ribert 1998] ou encore dans [Clavier 2000].

### 2.4.2.2. Les bases pour l'évaluation

Nous avons testé notre modèle sur 4 types de données différents :

- des données de synthèse, générées spécialement pour tester l'invariance à la rotation et au changement d'échelle. Ces données sont semblables à celles présentées en Figure 2.18, mais avec chaque lettre de l'alphabet en minuscule et en majuscule, ainsi que chaque chiffre de 0 à 9. Chaque caractère est décliné suivant 13 orientations et 2 tailles, ce qui engendre une base constituée de 1612 formes. Dans cette base, un certain nombre de formes sont similaires à une transformation géométrique près. Nous les avons regroupées dans des classes communes, constituant ainsi 54 classes. Sur le Tableau 2.3, nous représentons la répartition des différents prototypes en classes. Les regroupements de classes sont les suivants : {6-9} – {I-i-l} – {b-q} – {d-p} – {n-u} qui ont exactement la même forme dans la fonte utilisée. Notons enfin que pour les tests, cette base a été scindée aléatoirement en deux, une partie étant utilisée pour l'apprentissage, l'autre pour le test. Le programme utilisé pour le découpage de la base est aléatoire, mais il maintient la répartition par classes. Nous discuterons de la pertinence d'une telle approche en 2.4.5.

0	26	l	26	2	26	3	26	4	26	5	26	6	52	7	26	8	26
A	26	B	26	C	26	D	26	E	26	F	26	G	26	H	26	I	78
J	26	K	26	L	26	M	26	N	26	O	26	P	26	Q	26	R	26
S	26	T	26	U	26	V	26	W	26	X	52	Y	26	Z	52	a	26
C	26	e	26	f	26	g	26	h	26	J	26	k	26	m	26	n	52
O	26	p	52	q	52	r	26	s	26	T	26	v	26	w	26	y	26

*Tableau 2.3 : Nombre de prototypes par classe dans la base de synthèse*

- des données issues de la numérisation de 14 plans France Telecom. Dans le cadre du projet *NATALI*, nous avons été amené à numériser des plans pour tester nos différents outils. Ces plans, au format A0, ont été numérisés à une résolution de 400 **DPI (Dot Per Inch)**<sup>11</sup>. A l'issue de cette numérisation, nous avons procédé à une segmentation des composantes connexes, suivant des critères de taille comme énoncé en 2.1.3. 12 185 caractères ont ainsi été extraits, puis étiquetés. Parmi les classes présentes, certaines n'apparaissent que dans les noms de villes ou de rues, et étaient donc sous représentées par rapport à celles incluses dans le lexique du domaine. Nous avons

---

<sup>10</sup> <http://www.cis.hut.fi/research/som-research/nnrc-programs.shtml>

pris l'option de supprimer ces formes de notre base (en choisissant un seuil de suppression de 10 prototypes), afin de ne pas « perturber » le classifieur. A la suite de ce nettoyage de la base, nous disposons de 11 744 prototypes distribués en 33 classes comme le montre le Tableau 2.4. Là encore, nous avons regroupé les « 6 » et les « 9 » en une seule classe. Notons aussi que la classe « / » représente les tirets non supprimés par l'étape de segmentation, mais qui peuvent être éliminés ultérieurement par des traitements spécifiques. Nous avons utilisé la même technique pour la séparation base d'apprentissage / base de test que dans le cas des données de synthèse.

0	2118	l	1038	2	936	3	510	4	1016	5	1052	7	262	8	272	9	578
A	42	a	30	B	54	B	30	C	380	D	102	E	142	e	48	G	24
h	26	L	70	N	64	O	56	P	142	R	114	S	132	T	310	t	28
U	76	V	148	X	164	Z	12	(	38	/	1730						

*Tableau 2.4 : Nombre de prototypes par classe dans la base France Telecom*

- des données issues de la numérisation de 10 planches issues de la documentation EDF, fournies par Raphael Marc (EDF/DRD) dans le cadre d'une étude de faisabilité demandée à notre équipe. L'acquisition et les pré-traitements ont été effectués avec les mêmes outils que ceux utilisés pour les planches France Telecom. La base obtenue est constituée de 8158 prototypes répartis en 33 classes comme le montre le Tableau 2.5.

0	583	l	220	2	215	3	185	4	198	5	151	6	216	7	82	8	84
A	336	B	175	C	297	D	116	E	350	F	56	I	2244	L	139	M	88
N	162	P	209	R	142	S	247	T	318	U	234	V	85	X	292	Z	57
]	59	e	192	o	93	t	76	#	170	\$	87						

*Tableau 2.5 : Nombre de prototypes par classe dans la base EDF*

- des données issues de la base NIST de caractères manuscrits. Cette base est constituée de 60 000 caractères pour l'apprentissage, et de 10 000 pour le test. Les formes binaires constituant cette base ont d'abord été normalisées dans une taille 20\*20 puis insérées dans une imagerie 28\*28, le centre de masse de la forme étant au centre de l'imagerie. La répartition en classes de chacune des deux parties de la base est montrée dans le Tableau 2.6 et le Tableau 2.7. Dans les deux cas, nous avons supprimé de la base les caractères découpés en plusieurs parties. En effet, ce cas de figure ne se présentait pas dans le cadre de notre travail sur les documents techniques. Nous n'avons donc pas développé d'approche particulière.

---

<sup>11</sup> Point Par Pouce

0	5795	1	6387	2	5880	3	5960	4	5714	5	5222	6	5670	7	6104	8	5724
9	5690																

*Tableau 2.6 : Nombre de prototypes par classe dans la base d'apprentissage MNIST*

0	959	1	1090	2	1021	3	986	4	958	5	860	6	922	7	1008	8	947
9	976																

*Tableau 2.7 : Nombre de prototypes par classe dans la base de test MNIST*

### 2.4.2.3. Le paramétrage des modèles pour l'évaluation

Les différents modèles de formes utilisés pour cette évaluation sont tributaires d'un paramétrage de leur extracteur, qui influe sur la taille du vecteur correspondant. Nous reviendrons sur ce point, lors de la discussion et dans le cadre de la seconde partie du mémoire. Toutefois, nous citons ci-dessous la taille des vecteurs utilisés :

- Fourier Mellin : 33 invariants qui correspondent au paramétrage  $\sigma = 1$ ,  $r_{\max} = 20$ ,  $Q = 3$  et  $P = 2$
- Zernike : 24 moments qui correspondent aux moments jusqu'à l'ordre 6.
- Sondes circulaires : 40 sondes qui correspondent à 4 cercles concentriques superposés à la forme. Sur chacun de ces cercles 10 angles sont calculés (cf. 2.2.3.7 et Figure 2.8).

### 2.4.2.4. Résultats sur la base de synthèse

#### *i. Avec le classifieur LVQ*

Les résultats obtenus par le classifieur LVQ sur la base de synthèse sont donnés dans le Tableau 2.8 sous la forme de taux de reconnaissance par classe :

0	84,6%	1	100%	2	100%	3	100%	4	100%	5	100%	6	100%	7	100%	8	100%
A	100%	B	100%	C	100%	D	84,6%	E	100%	F	100%	G	100%	H	100%	I	76,9%
J	100%	K	100%	L	100%	M	100%	N	100%	O	76,9%	P	100%	Q	76,9%	R	100%
S	92,3%	T	100%	U	100%	V	100%	W	84,6%	X	100%	Y	100%	Z	84,6%	a	100%
C	100%	e	92,3%	f	92,3%	G	100%	h	100%	J	100%	k	100%	m	84,6%	n	100%
O	84,6%	p	100%	q	100%	R	100%	s	100%	T	100%	v	100%	w	100%	y	100%

*Tableau 2.8 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base de synthèse*

Ces résultats correspondent à un taux de reconnaissance global de 97%. Les erreurs principales sont commises sur les classes « 0 », « Q » et « O ». Cela est dû à des confusions logiques entre ces trois classes, les formes étant relativement proches. Nous avons comparé ce résultat avec les deux modèles

de la littérature que nous avons présentés précédemment, les résultats globaux sont donnés par le Tableau 2.9 :

Modèle de représentation	Taux de reconnaissance
<i>Fourier Mellin</i>	97 %
<i>Zernike</i>	75,5 %
<i>Sondes Circulaires</i>	73,6 %

Tableau 2.9 : Comparaison entre les modèles sur la base de synthèse avec le LVQ

Ce tableau met nettement notre modèle en avant. Cependant, il faut signaler qu'une telle différence n'est pas uniquement liée à la qualité du modèle, mais aussi à des problèmes de dimensionnalité. Nous reviendrons sur ces considérations dans la discussion (2.4.5).

**ii. Avec le classifieur KPPV**

Nous avons effectué rigoureusement les mêmes tests en utilisant un classifieur de type 1-ppv. Le taux de reconnaissance obtenu est légèrement supérieur à celui donné par le LVQ (97,5%), ce qui est en adéquation avec les résultats généralement exposés dans la littérature. La matrice de confusion sur un ensemble de 54 classes (tableau de 54 colonnes et 54 lignes) étant difficile à montrer pour des questions de lisibilité et les confusions étant très rares, nous présentons plutôt dans Tableau 2.10 l'ensemble des confusions obtenues sous forme de liste. Ensuite, le Tableau 2.11 présente le comparatif entre les différents modèles en utilisant l'approche 1-Plus proche voisin pour la classification.

<i>Classe Vraie</i>	0	0	O	O	Q	Q	8	M	W	V	N	e	J	I
<i>Classe Reconnue</i>	O	Q	0	Q	O	0	B	W	M	U	Z	a	I	J
<i>Nombre Occurrences</i>	1	1	2	1	1	2	2	2	1	1	2	1	1	2

Tableau 2.10 : Les confusions du test du 1-PPV avec le modèle de FM

Modèle de représentation	Taux de reconnaissance
<i>Fourier Mellin</i>	97,5 %
<i>Zernike</i>	80,9 %
<i>Sondes Circulaires</i>	77,5 %

Tableau 2.11 : Comparaison entre les modèles sur la base de synthèse avec le 1-PPV

**2.4.2.5. Résultats sur la base France Telecom**

Dans cette partie, nous présentons l'ensemble des résultats que nous avons obtenus avec la base constituée de caractères issus de 14 plans France Telecom. Nous présentons d'abord les résultats obtenus avec le LVQ, puis avec les Kppv. Dans ce cadre, nous montrons également quelques résultats obtenus en faisant varier le paramètre k.

*i. Avec le classifieur LVQ*

Sur la base France Telecom, un taux de reconnaissance global de reconnaissance de 95,1% est obtenu. Le Tableau 2.12 montre ce résultat exprimé par classe.

0	99,7%	1	98,0%	2	99,6%	3	98,8%	4	98,4%	5	97,7%	7	90,8%	8	94,8%	9	92,7%
A	61,9%	a	46,7%	B	40,7%	b	46,7%	C	97,9%	D	52,9%	E	88,7%	e	95,8%	G	91,7%
h	30,8%	L	88,6%	N	43,8%	O	75,0%	P	97,2%	R	87,7%	S	63,6%	T	94,8%	t	42,9%
U	65,8%	V	97,3%	X	98,8%	Z	00,0%	(	15,8%	/	99,33						

*Tableau 2.12 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base FT*

Dans ce tableau, il peut être remarqué que les résultats ne sont pas homogènes, les taux de reconnaissance variant considérablement d'une classe à une autre. Selon nous, cela n'est pas dû à la qualité du modèle utilisé mais à l'hétérogénéité des classes qui perturbe le processus de classification. Les classes les plus représentées sont bien reconnues alors que les classes « pauvres » en prototypes obtiennent de mauvais taux. A titre d'illustration, en ne prenant en compte que les chiffres qui sont majoritaires sur nos documents, une base de 7226 caractères est obtenue. Pour cette base, le taux de reconnaissance global augmente jusqu'à 98%. Cela nous rend optimiste quant aux scores envisageables dans le cadre d'une constitution « industrielle » de bases représentatives.

Avec cette base et ce classifieur, nous avons comparé nos différents modèles. Là encore, la supériorité du modèle de Fourier Mellin est mise en avant (Tableau 2.13) :

<b>Modèle de représentation</b>	<b>Taux de reconnaissance</b>
<i>Fourier Mellin</i>	95,1 %
<i>Zernike</i>	83,9 %
<i>Sondes Circulaires</i>	82,6 %

*Tableau 2.13 : Comparaison entre les modèles sur la base France Telecom avec le LVQ*

*ii. Avec le classifieur KPPV*

Avec la même base, le classifieur 1-ppv donne de nouveau un résultat légèrement supérieur. Le taux de reconnaissance atteint 95,8%. Le Tableau 2.14 présente les taux de reconnaissance par classe obtenus pour ce test. Les résultats sont similaires à ceux obtenus avec le LVQ. Les mêmes classes, souvent celles pauvres en prototypes, posent des problèmes alors que les classes très représentées sont bien reconnues.

Afin de montrer les erreurs commises par ce test, nous présentons sur le Tableau 2.15 la matrice de confusion correspondant à ce test. Elle permet de mettre en avant les plus grosses confusions qui sont



explicables par la similarité des formes (1-7 ; 5-9 ; D-0 ;...). Notons que ces confusions sont les mêmes lorsque le LVQ est utilisé.

0	99,6%	1	97,1%	2	99,4%	3	98,4%	4	98,2%	5	97,5%	7	88,5%	8	96,3%	9	94,5%
A	61,9%	a	80%	B	66,7%	b	60%	C	96,3%	D	70,6%	E	95,8%	e	95,8%	G	100%
h	61,5%	L	91,4%	N	56,3%	O	60,7%	P	98,6%	R	93%	S	84,8%	T	94,2%	t	57,1%
U	71,1%	V	97,3%	X	97,6%	Z	66,7%	(	47,4%	/	99,8%						

Tableau 2.14 : Taux de reconnaissance par classe obtenus par FM et 1 ppv sur la base FT

)	/	0	1	2	3	4	5	7	8	9	A	B	C	D	E	G	L	N	O	P	R	S	T	U	V	X	Z	a	b	e	h	t		
)	9	0	0	4	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		
/	5	863	0	2	1	0	0	0	0	0	0	1	1	0	0	0	0	5	3	0	0	0	5	0	0	1	0	0	0	0	0	1		
0	0	0	1055	0	0	0	0	0	0	0	0	1	1	11	0	0	0	0	6	0	0	0	0	4	0	0	0	0	0	0	0	0		
1	1	1	0	504	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1		
2	0	0	0	0	465	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0		
3	0	0	0	0	0	251	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0		
4	0	1	0	0	0	1	499	1	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	1	513	0	0	13	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	
7	0	0	0	8	1	0	4	0	116	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
8	0	0	1	0	0	0	0	0	0	131	1	0	4	0	0	1	0	0	1	0	0	3	0	0	0	0	0	0	0	0	0	1	0	0
9	0	0	0	0	0	0	8	0	0	273	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	
A	0	0	0	0	0	0	0	0	0	0	13	0	0	0	0	0	0	0	0	0	0	0	0	2	1	0	0	1	0	0	0	0	0	
B	0	0	0	0	0	0	0	0	1	0	0	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	
C	2	0	0	0	0	0	0	0	0	0	0	0	183	1	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	
D	0	0	2	0	0	0	0	0	0	0	0	0	0	0	36	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	
E	0	0	0	0	0	0	0	0	0	0	0	0	0	0	68	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
G	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
L	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	32	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
N	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	
O	0	0	1	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
P	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	70	0	0	0	0	0	0	0	0	0	0	0	0	4	0
R	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	53	1	0	0	0	0	0	0	0	0	0	0	0	0
S	2	0	0	0	1	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	56	0	0	0	0	0	0	0	0	0	0	0	0	
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	146	0	0	0	0	0	0	0	0	0	0	4	0
U	0	0	0	0	0	1	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	27	0	0	0	0	0	0	0	0	0	0	0
V	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	72	0	0	0	0	0	0	0	0	0	0
X	0	0	0	0	0	3	0	0	0	3	0	0	0	0	0	0	0	2	0	0	1	4	0	0	0	80	2	0	0	0	0	0	0	
Z	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
a	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	
b	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	0	1	0	0	0	
e	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	23	0	0	0	0	
h	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0
T	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	8	0

Tableau 2.15 : Matrice de confusion du test du 1-PPV avec le modèle de FM

Comme dans le cas précédent, nous avons comparé ces résultats aux modèles issus de la littérature ; les résultats de cette comparaison sont présentés dans le Tableau 2.16. Les résultats sont équivalents à ceux obtenus par le LVQ, c'est à dire une nette supériorité du modèle de Fourier Mellin sur les deux autres.

Modèle de représentation	Taux de reconnaissance
Fourier Mellin	95,8 %
Zernike	83,5 %
Sondes Circulaires	83,9 %

Tableau 2.16 : Comparaison entre les modèles sur la base France Telecom avec 1 ppv

Dans le cadre de ces tests associant le kppv avec le modèle de Fourier Mellin sur la base issue des documents France Telecom, nous avons également effectué des tests de variation du paramètre k. En effet, dans le cadre d'un projet industriel comme le nôtre, l'objectif est d'obtenir un bon taux de reconnaissance, mais surtout de réduire au maximum le taux de confusion. Prendre un k différent de 1 nous permet d'introduire du rejet dans notre processus de classification, en utilisant la règle du vote majoritaire. Les résultats obtenus sont présentés Figure 2.24 par l'intermédiaire d'une courbe exprimant le taux de reconnaissance (ou taux de lecture) par rapport au taux de confusion. Cette courbe a pour but de déterminer la valeur de k adéquate, en fonction du compromis taux de reconnaissance / taux de confusion désiré. Par exemple, si le but est d'avoir un taux d'erreur inférieur à 1%, le meilleur choix est d'utiliser 6 voisins, qui permettent l'obtention de 0,98% d'erreur, pour 84,11 % de reconnaissance. En terme de comparaison, la supériorité du modèle de Fourier Mellin est de nouveau mise en avant sur cette courbe, quel que soit le nombre de voisins utilisé. De plus, mis à part le cas du 1ppv, le modèle de Zernike se révèle légèrement plus efficace que le modèle basé sur les sondes circulaires. Néanmoins, dans les deux cas, l'obtention d'un taux d'erreur inférieur à 1% implique un taux de reconnaissance inférieur à 50%, ce qui constitue une différence supérieure à 30 % par rapport au modèle de Fourier Mellin.

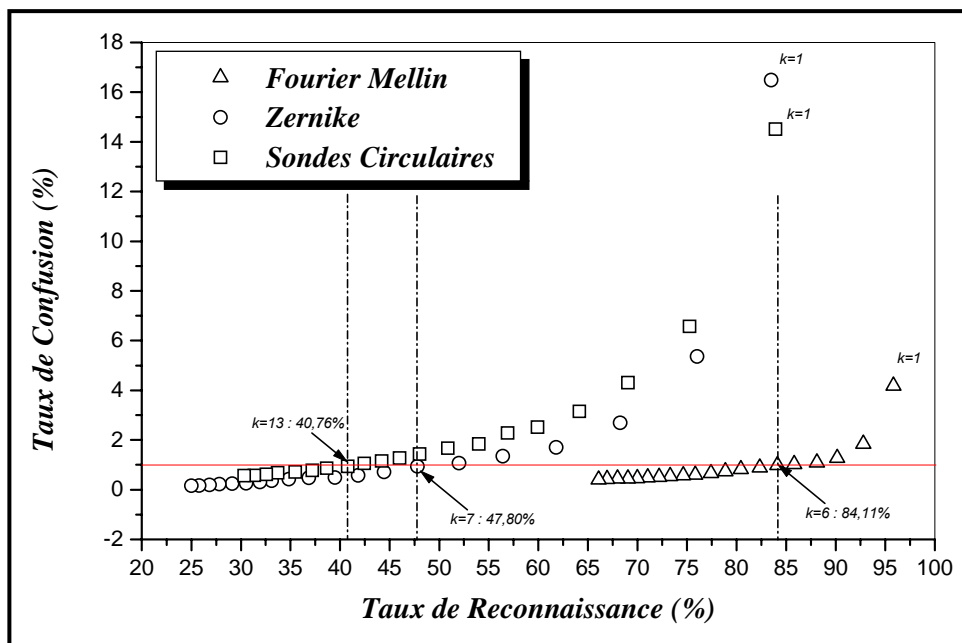


Figure 2.24 : Courbe taux de confusion / taux de lecture en faisant varier le nombre de voisins

#### 2.4.2.6. Résultats sur une base EDF

Nous avons également testé le modèle de Fourier Mellin sur des données issues de plans de réseau électrique [Barbey 1999]. Dans le Tableau 2.17 nous présentons les résultats obtenus par le LVQ avec cette base EDF. Le taux de reconnaissance global atteint alors 94,36 %.

0	96,7%	1	86,4%	2	93,5%	3	98,4%	4	96,5%	5	93,4%	6	96,8%	7	71,9%	8	79,8%
A	96,7%	B	92,6%	C	98,7%	D	52,6%	E	98,0%	F	85,7%	I	99,7%	L	84,2%	M	85,2%
N	96,9%	P	94,3%	R	92,9%	S	87,9%	T	90,9%	U	89,7%	V	84,7%	X	97,9%	Z	71,9%
]	91,5%	E	95,8%	o	84,9%	t	94,7%	#	91,2%	\$	98,9%						

Tableau 2.17 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base EDF

Dans ce tableau, les résultats sont à peu près similaires à ceux obtenus avec la base France Telecom. Le fait qu'ils soient légèrement inférieurs s'explique parfaitement par le fait que la base soit formée de moins de prototypes, avec un nombre de classes supérieur.

#### 2.4.2.7. Résultats sur une base NIST

Pour valider définitivement notre approche, nous avons effectué quelques tests sur une partie de la base NIST de caractères manuscrits. L'objectif de ces tests n'était pas de rivaliser avec les approches classiques, qui ne prennent pas en compte une éventuelle multi-orientation des formes (absente dans la base NIST) mais plutôt de comparer les résultats entre les différents modèles. Les tableaux suivants donnent les taux de reconnaissance par classe avec le LVQ et les trois modèles différents. Le Tableau 2.21 propose une comparaison entre les taux de reconnaissance obtenus par les différents modèles.

0	97,1%	1	99,4%	2	95,9%	3	93,8%	4	91,6%	5	92,9%	6	82,3%	7	97,6%	8	95,1%
9	86,7%																

Tableau 2.18 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur NIST

0	90,0%	1	92,8%	2	89,9%	3	87,3%	4	87,8%	5	90,1%	6	72,9%	7	92,7%	8	90,1%
9	75,2%																

Tableau 2.19 : Taux de reconnaissance par classe obtenus par Zernike et le LVQ sur NIST

0	91,5%	1	91,9%	2	90,8%	3	87,1%	4	86,9%	5	91,2%	6	74,7%	7	91,3%	8	89,8%
9	76,7%																

Tableau 2.20 : Taux de reconnaissance par classe obtenus par les sondes circulaires et le LVQ sur NIST

Modèle de représentation	Taux de reconnaissance
<i>Fourier Mellin</i>	90,84 %
<i>Zernike</i>	84,63 %
<i>Sondes Circulaires</i>	84,90 %

Tableau 2.21 : Comparaison entre les modèles sur la base NIST avec le LVQ

Ces tableaux montrent que notre approche obtient des résultats corrects, malgré la forte confusion entre les classes « 6 » et « 9 ». Cette confusion est tout à fait logique puisque les modèles sont invariants au changement d'orientation. Toutefois, il faut signaler que bon nombre d'OCR de la

littérature obtiennent des taux supérieurs de plusieurs points, même pour les classes bien reconnues par notre système.

### 2.4.3. Les résultats obtenus pour les formes connectées

#### 2.4.3.1. Résultats sur des formes synthétiques

Pour évaluer l'outil de reconnaissance de formes connectées, nous avons d'abord voulu valider d'un point de vue conceptuel cette approche, en prouvant son apport par rapport à la technique utilisée pour traiter les formes isolées. Afin de mettre en évidence la fiabilité de ce procédé de filtrage, au regard de la connexion de différents objets, nous avons testé les deux stratégies sur un jeu de formes simples, constituées d'un « 3 » tourné, progressivement perturbé par l'introduction d'une barre verticale, comme le montre la Figure 2.25. Sous les images des « 3 » figurent les résultats donnés par les deux techniques. Celui issu d'une approche « filtrage » est à gauche, avec une confiance associée alors que le résultat obtenu avec l'approche classique est entre parenthèses. Hormis sur la 3<sup>ème</sup> ligne, aucune erreur n'est commise avec l'approche à base de filtrage alors qu'en exploitant l'approche « composantes connexes », plusieurs erreurs sont commises. Ces résultats nous ont permis de poursuivre plus avant ces travaux, et nous ont ensuite amené à tester l'approche sur les données France Telecom.

3 0.92  (3)	3 0.75  (3)	3 0.76  (3)	3 0.78  (3)	3 0.80  (3)	3 0.82  (3)	3 0.83  (3)	3 0.83  (3)	3 0.83  (3)	3 0.82  (3)
3 0.83  (3)	3 0.82  (3)	3 0.79  (3)	3 0.79  (3)	3 0.77  (3)	3 0.77  (3)	3 0.76  (3)	3 0.75  (3)	3 0.73  (3)	3 0.71  (3)
3 0.71  (9)	3 0.71  (9)	3 0.71  (9)	3 0.70  (8)	3 0.67  (8)	3 0.67  (8)	3 0.78  (8)	3 0.79  (8)	3 0.79  (6)	3 0.78  (6)
3 0.58  (8)	3 0.59  (8)	3 0.64  (8)	3 0.66  (8)	3 0.69  (0)	3 0.70  (0)	3 0.73  (0)	3 0.73  (0)	3 0.75  (0)	3 0.69  (0)
3 0.69  (0)	3 0.76  (0)	3 0.80  (3)	3 0.80  (3)	3 0.92  (3)					

Figure 2.25 : Illustration des résultats du filtrage sur des formes de synthèse

#### 2.4.3.2. Résultats sur les données France Telecom

Pour les formes connectées des documents France Telecom, la stratégie que nous avons appliquée était la suivante. Tout d'abord, un ensemble de 257 sous-images comprenant des caractères connectés a été

sélectionné manuellement. Sur ces images, les formes collées pouvaient être des caractères connectés les uns aux autres ou des caractères connectés au réseau (exemple Figure 2.26). Un nombre total de 408 caractères collés était présent sur l'ensemble des images. Les résultats obtenus sont très bons puisque le taux de classification correcte atteint 83%. Comme dans le cas des formes synthétiques, les principales erreurs de classification sont essentiellement dues aux formes semblables. Cependant, dans certains cas particuliers, malgré la faible influence des formes connectées sur la description des formes à reconnaître, certaines erreurs semblent dues à une légère perturbation du calcul des invariants. Sur la Figure 2.27, nous présentons une illustration de l'application du mode filtrage permettant la détection (Figure 2.27, image centrale) et la reconnaissance d'un caractère "3" (Figure 2.27, image de gauche) initialement connecté au réseau. Notons qu'avec l'approche utilisée, nous ne caractérisons pas la capacité de détection de notre outil, mais uniquement la capacité de reconnaissance d'une forme dans une zone pré-détectée.



Figure 2.26 : Exemples de caractères collés sur les plans France Telecom

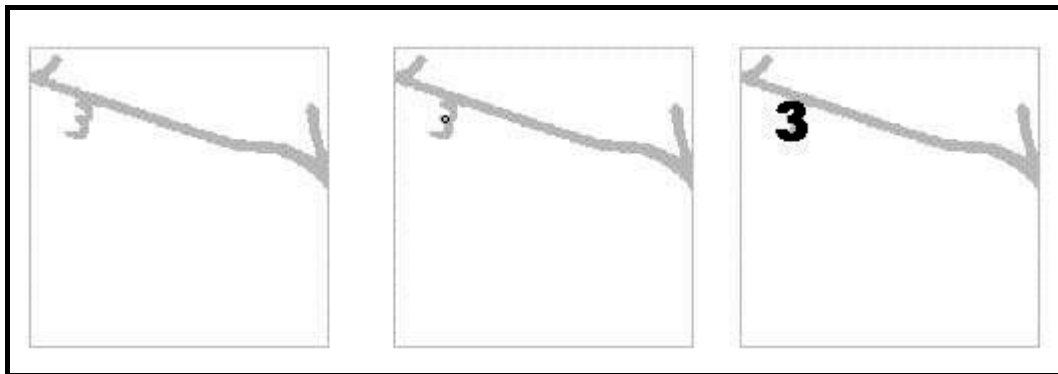


Figure 2.27 : Illustration du traitement des formes collées

Ces premiers résultats, bien qu'insuffisants, sont encourageants. Il est toutefois important de noter que compte tenu de l'importante complexité algorithmique d'une telle approche, il est nécessaire d'adopter un mécanisme permettant de détecter des zones d'intérêt dans lesquelles appliquer ce « filtrage ». Nous reviendrons sur ce point dans la conclusion générale de ce mémoire.

## 2.4.4. Les résultats pour l'estimation des paramètres de mouvement

### 2.4.4.1. Méthodologie d'évaluation

Dans la partie 2.3.5, nous avons montré qu'il était possible d'exploiter la transformée de Fourier Mellin pour estimer l'orientation et l'échelle d'une forme préalablement reconnue, en utilisant une

base de descripteurs issue de formes droites et l'équation (96). L'algorithme d'application d'une telle théorie a ensuite été montré en 2.3.6.3, ainsi que quelques illustrations en 2.3.7.3. Dans cette partie, nous présentons quelques résultats quantitatifs concernant cette estimation des paramètres de mouvement des formes. La méthodologie utilisée a consisté à construire une base d'images de formes droites, à en extraire les descripteurs de Fourier Mellin et à les étiqueter par la classe de la forme (ce qui explique la relative pauvreté de la base). A partir de cette base, nous avons testé l'approche sur les mêmes formes que nous avons tournées par un algorithme simple. Les résultats sont présentés ci-dessous, en utilisant deux mécanismes différents d'optimisation, à savoir l'algorithme de Nelder Mead et les algorithmes génétiques. Ensuite, nous présentons les résultats issus d'une comparaison entre notre approche et une approche concurrente, issue de travaux de Kim [Kim 1999]. Cette méthode utilise la même philosophie exploitant la complétude de la transformation de l'image, mais à partir des moments de Zernike.

#### **2.4.4.2. Les résultats : comparaison entre les 2 méthodes d'optimisation**

Dans le Tableau 2.22, nous montrons quelques résultats concernant l'estimation des deux paramètres de mouvement des formes (dans les deux dernières colonnes, le facteur de taille estimé est exprimé à gauche, l'estimation de l'orientation est donnée en degrés à droite) en utilisant les deux méthodes d'optimisation présentées dans 2.3.6.3

<i>Caractère</i>	<i>Facteur de Taille</i>	<i>Orientation</i>	<i>Nelder Mead</i>	<i>Algorithmes Génétiques</i>
<i>E</i>	1	0	1 / 9.8 E-6	0.99 / 359
<i>A</i>	1	0	1 / -1.26 E-5	0.95 / 1.47
<i>C</i>	1	0	1 / -2.77 E-7	1.00 / 2.72
<i>E</i>	0.5	15	0.49 / 12.15	0.49 / 12.72
<i>A</i>	1	45	1.01 / 46.56	1.01 / 47.23
<i>A</i>	1	60	0.99 / 61.40	1.02 / 63.83
<i>C</i>	1	90	0.98 / 93.74	0.98 / 93.32
<i>A</i>	0.5	90	0.51 / 90.74	0.51 / 90.78

*Tableau 2.22 : L'estimation du facteur d'échelle et de l'orientation*

Ici, les résultats obtenus avec les deux méthodes d'optimisation sont proches. Tous les résultats présentés ici sont corrects. Toutefois, en terme de temps de calcul, la méthode basée sur les algorithmes génétiques est beaucoup plus rapide (facteur 2 à 3) comme nous l'avions présupposé.

#### **2.4.4.3. Les résultats : comparaison entre Fourier Mellin et Zernike**

Le Tableau 2.23 présente une comparaison entre les résultats fournis par notre méthode et ceux obtenus par l'approche issue de travaux de Kim [Kim 1999]. Avec la méthode basée sur la transformée de Fourier Mellin l'erreur moyenne est de 14,41 degrés alors que l'approche basée sur les moments de Zernike fournit une erreur de 12,32 degrés. Ce taux élevé est dû à la présence de quelques grosses

erreurs, qui perturbent évidemment la moyenne finale. De plus, certaines de ces erreurs sont selon nous dues à l'algorithme de rotation que nous avons utilisé, qui produit des formes très perturbées. Concernant le choix du « meilleur » outil, si l'approche basée sur les moments de Zernike semble supérieure, il faut prendre en compte le fait que cette estimation est contrainte par la reconnaissance. L'estimation des paramètres de mouvement ne peut se faire dans les deux cas que si la forme a correctement été reconnue. Le modèle de Fourier Mellin étant supérieur au modèle de Zernike, il semble plus intéressant pour l'estimation des paramètres de mouvement.

<b>Orientation</b>	<b>Méthode</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>
0	Fourier-Mellin	0	0	0	0	0
0	Zernike	0	0	0	0	0
15	Fourier-Mellin	16	15	17	13	17
15	Zernike	16	16	14	22	290
30	Fourier-Mellin	30	30	32	33	28
30	Zernike	23	282	33	30	34
45	Fourier-Mellin	47	-154	47	47	44
45	Zernike	48	144	46	47	47
60	Fourier-Mellin	61	60	67	62	63
60	Zernike	61	44	331	55	60
75	Fourier-Mellin	78	73	78	77	76
75	Zernike	76	74	77	75	76
90	Fourier-Mellin	91	89	94	90	94
90	Zernike	88	82	89	92	92
105	Fourier-Mellin	105	-84	108	106	101
105	Zernike	107	103	106	105	108
120	Fourier-Mellin	118	-50	121	119	119
120	Zernike	122	121	40	120	199
135	Fourier-Mellin	136	136	134	135	136
135	Zernike	137	134	40	138	138
150	Fourier-Mellin	-48	148	150	149	152
150	Zernike	153	152	63	151	151
165	Fourier-Mellin	164	166	163	167	169
165	Zernike	167	167	163	165	167
180	Fourier-Mellin	-15	172	174	177	176
180	Zernike	175	184	177	177	176

*Tableau 2.23 : L'estimation de l'orientation : comparaison entre Fourier Mellin et Zernike*

### **2.4.5. Conclusion et discussion sur l'évaluation**

Dans ce paragraphe, de très nombreux résultats quantitatifs ont été exposés et commentés. La méthodologie utilisée pour cette évaluation a été exposée dans l'introduction du chapitre. Elle constitue un choix qui est à de nombreux points de vue contestable. C'est pourquoi dans cette section, nous proposons de mettre en avant les points que nous n'avons pas abordés, pour des questions de temps, mais qui méritent d'être évoqués et qui constituent des perspectives pour la suite de ce travail. Nous regroupons ces remarques suivant 3 points : la constitution des bases pour les tests, le paramétrage des modèles et le paramétrage des classifieurs.

- La constitution des bases : Pour chacune des bases utilisées dans l'évaluation, la base originale a été aléatoirement séparée en deux parties égales, l'une pour l'apprentissage, l'autre pour le test, en respectant la répartition du nombre de prototypes par classes. Une telle approche est évidemment contestable puisque les résultats sont soumis au hasard de la séparation. Afin de tester la sensibilité des résultats au tirage aléatoire, nous avons au préalable effectué quelques tests en répétant plusieurs fois (5) le tirage et en calculant la moyenne et la variance obtenues. Les résultats de ces tests ont montré une très faible variabilité des résultats. C'est pourquoi nous avons choisi de ne plus utiliser une telle approche, coûteuse en temps de calcul, et de ne faire ce tirage qu'une fois. Néanmoins, signalons qu'il aurait été intéressant de comparer différentes stratégies d'évaluation des résultats, en considérant par exemple la méthode de « cross validation » [Kearns 1997].
- Le paramétrage des modèles : nous avons présenté en 2.4.2.3 les choix de paramétrage de nos extracteurs de modèles de formes. Ce problème de paramétrage, qui induit directement celui du dimensionnement de l'espace de représentation est un problème récurrent pour la communauté scientifique du domaine. Ainsi, de très nombreux travaux concernent la recherche de règles déterminant des ratios idéaux entre taille des bases, taille de l'espace de représentation et nombre de classes. Néanmoins, à notre connaissance, aucune étude n'a réellement permis de déterminer de telles règles. En menant quelques tests afin de choisir un espace de représentation optimisant le taux de reconnaissance pour notre problème, sur la base des données France Telecom, nous avons pu constater la grande influence de ce choix. Ces travaux consistaient à optimiser l'espace de représentation en sélectionnant les primitives pertinentes à l'aide des algorithmes génétiques [Adam 2000b] (cf. 2.3.6.3 i.). Les résultats obtenus, dont une illustration est donnée sur la Figure 2.28, valident les remarques effectuées ci-avant. La courbe présentée correspond à l'évolution du score (en l'occurrence le taux de reconnaissance) du meilleur individu de la population au fur et à mesure des générations. Cette courbe montre qu'en fonction du jeu de primitives choisi parmi les trois modèles développés, le taux de reconnaissance peut varier de plus de 20 % d'un espace à un autre. Elle montre également l'efficacité des algorithmes génétiques pour optimiser un problème



tel que celui de la sélection de primitives. Les travaux concernant cette approche sont actuellement poursuivis mais n'entrent pas réellement dans le cadre de cette partie de la thèse puisqu'ils ne concernent pas directement l'évaluation d'un modèle. Néanmoins, les évoquer ici permet d'illustrer les propos concernant l'importance de l'espace de représentation choisi.

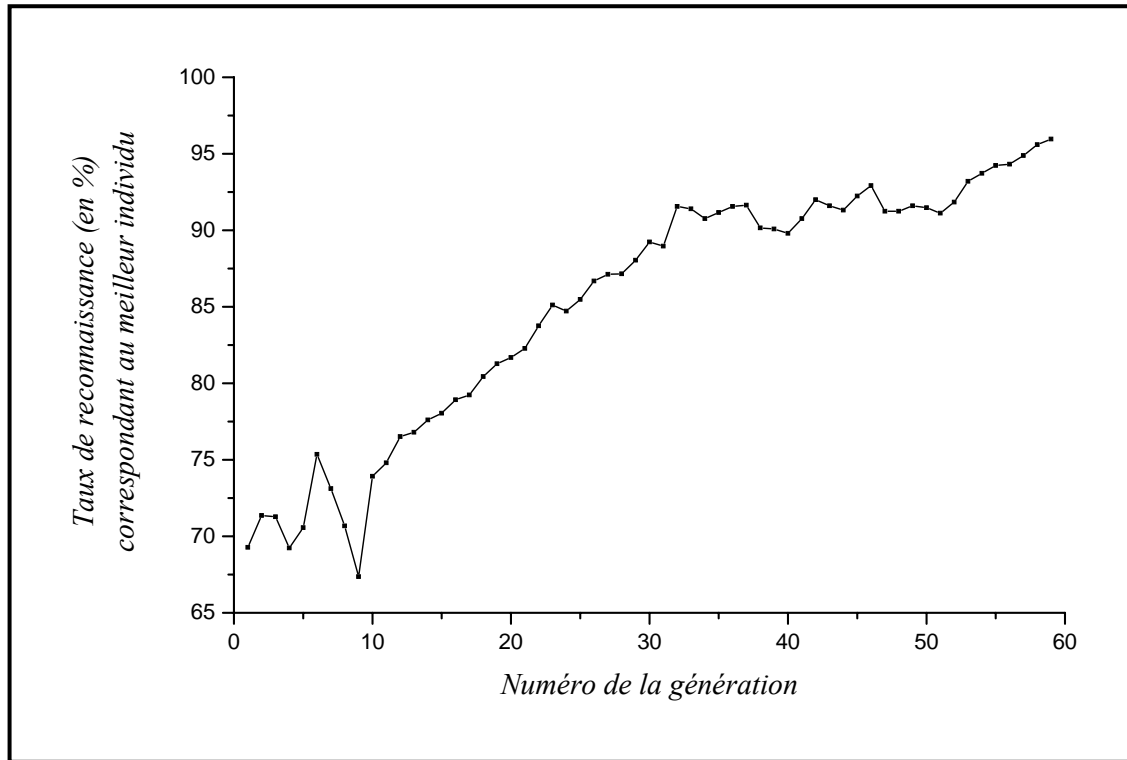


Figure 2.28 : Evolution du score du meilleur individu lors du déroulement de l'algorithme génétique

- Le paramétrage des classifieurs. Si le choix d'un modèle de représentation des formes est incontestablement une clef de voûte d'un système de reconnaissance, la partie qui permet de prendre la décision (la classification dans notre cas puisque nous travaillons avec des vecteurs de caractéristiques) revêt-elle aussi une grande importance. Nous avons évoqué lors de la présentation succincte des classifieurs les nombreux paramètres qui doivent être réglés pour les utiliser. Pour le kppv, le choix de la distance, du nombre de voisins et de la méthode de décision peut faire varier les résultats de plusieurs points. Dans le cas du LVQ, le nombre de neurones, de cycles d'apprentissage et la fonction de base sont également des choix influant grandement sur les performances du système. Faire varier tous ces paramètres pour en optimiser les valeurs est un travail fastidieux, et qui au final, sera valable uniquement sur le problème ou la base ayant permis l'optimisation. Nous n'avons donc pas cherché à régler de façon optimale tous les paramètres. Ce point relève en effet selon nous plutôt du système global. Les réglages devront plutôt être effectués au fur et à mesure de la vie du système par apprentissage. Ce point sera au centre de la seconde partie du manuscrit.

## **2.5. Conclusion**

Dans cette première partie de la thèse, nous avons abordé le problème de la reconnaissance de forme d'un point de vue « bas niveau ». Après avoir présenté notre modèle de stratégie de reconnaissance d'une forme, basé sur le passage par un espace de représentation, puis sur l'appariement entre cette représentation et les éléments d'une base d'apprentissage, nous nous sommes focalisé sur l'extraction de modèles de représentation. Dans ce cadre, nous avons présenté un ensemble de modèles proposés par la littérature, en articulant cette synthèse en fonction de l'information initiale qui est utilisée pour l'extraction. Dans cette partie bibliographique, nous avons mis l'accent sur les approches respectant les contraintes de notre étude. Deux principales conclusions ont été tirées : d'une part la supériorité reconnue des moments de Zernike pour modéliser les formes de façon invariante, mais aussi le manque de modèle permettant d'appréhender les formes connectées.

A la suite de cette étude bibliographique, nous avons présenté un nouveau modèle de représentation de formes basé sur la transformée de Fourier Mellin. Après un rapide rappel de quelques notions mathématiques, nous avons présenté cette transformée d'un point de vue théorique et exposé des propriétés déduites de cette théorie, et intéressantes au regard de notre problème. Nous avons montré comment appliquer cette transformée à la structure bi-dimensionnelle discrète caractéristique des images numériques. Les algorithmes d'application de ce principe ont été donnés, ainsi que quelques illustrations visuelles des propriétés. Dans le chapitre suivant, nous avons exposé l'ensemble des résultats obtenus au cours de ce travail pour les trois propriétés intéressantes de la TFM, à savoir :

- La reconnaissance des formes isolées multi-orientées et multi-échelles : en associant un classifieur au modèle de Fourier Mellin, nous avons montré que celui-ci se révèle très efficace pour reconnaître les formes présentes sur des documents techniques. En le comparant à deux modèles reconnus efficaces par la littérature sur différentes bases, nous avons mis en avant la puissance d'un tel modèle. Dans la partie suivante, nous montrerons que ce modèle, et les bons taux de reconnaissance qui découlent de son utilisation, occupent une place centrale dans notre système d'interprétation
- La reconnaissance des formes connectées : le modèle de Fourier Mellin est particulièrement efficace pour modéliser des formes lorsque celles-ci sont connectées. Nous avons montré des résultats probants sur des images synthétiques, ainsi que sur quelques données issues de plans France Telecom.
- L'estimation des paramètres de mouvement des formes : en disposant de descripteurs issus de formes dans une position de référence, nous avons montré par l'intermédiaire de deux tableaux

qu'il était possible d'estimer les paramètres de mouvement des formes. Cette information est très intéressante pour la reconstruction de chaînes de caractères cohérentes.

Une discussion sur l'ensemble des résultats présentés a enfin permis de montrer les lacunes de notre approche en terme d'évaluations. Celle-ci a ouvert de nombreuses perspectives pour de nouveaux tests de validation. L'ensemble des perspectives de poursuite de ces travaux sera exposé dans la conclusion générale de ce mémoire.

A la suite de cette partie « Outils pour la reconnaissance de textes de symboles », nous allons, dans la seconde partie de cette thèse, aborder le sujet de leur intégration dans un système global d'interprétation de documents techniques.

---

# **Partie III : Intégration des Outils dans un Système à Base de Connaissances**

- 3.1. Introduction**
- 3.2. Les connaissances d'un système d'interprétation**
- 3.3. Les connaissances dans la littérature du domaine**
- 3.4. L'expérience de *NATALI v1***
- 3.5. De l'analyse critique à *NATALI v2***
- 3.6. Conclusion**

### **3. Intégration des outils dans un système à base de connaissances**

#### **3.1. Introduction**

Dans l'introduction générale de ce mémoire, une définition d'un système de rétro conversion de documents techniques a été proposée. Un tel système vise à extraire, le plus « intelligemment » possible, tout ou partie des informations métier présentes sur le document papier, en fonction des intentions de l'utilisateur du système. Un processus d'interprétation peut donc être assimilé à une tâche d'instanciation progressive d'un modèle « objets métier » du document ou d'une partie de celui-ci. Cette instanciation est réalisée en utilisant des procédures de traitement d'images et de reconnaissance de formes, telles que celles que nous avons décrites dans la partie précédente. Ces traitements sont guidés par une stratégie d'interprétation. Celle-ci doit être déduite de la requête émise par l'utilisateur du système, des connaissances du système et d'éventuelles coopérations avec l'utilisateur. Elle est alors appliquée par un système logiciel permettant l'ordonnancement des traitements, le transfert de données entre ces traitements, l'Interaction **H**omme **M**achine (**I.H.M.**), et éventuellement l'archivage de ces informations dans une structure de stockage adaptée à l'organisation des données métier.

Cette définition montre qu'un système d'interprétation de documents techniques ou plus généralement d'analyse de scènes, doit disposer d'une grande quantité de connaissances diverses. Parmi celles-ci figurent évidemment celles de l'expert du domaine considéré par le document ou par la scène (concepteur du réseau, photo interprète, médecins...). Celui-ci est en effet capable, en se basant sur son expertise, de définir le modèle des données à extraire du point de vue de sa spécialité. Les connaissances du traiteur d'images sont elles aussi fondamentales. Elles permettent d'une part de décrire les outils nécessaires à l'extraction des informations contenues dans l'image, et d'autre part d'exprimer la façon d'utiliser ces outils. Les connaissances d'un spécialiste de génie logiciel sont également primordiales. Elles permettent la construction d'un système informatique complexe et fiable, afin d'en assurer la réutilisabilité, la maintenance et l'évolution.

Un système d'interprétation de documents techniques, comme d'ailleurs tout système d'interprétation au sens large du terme (incluant les systèmes de traitement de la langue naturelle par exemple), est donc clairement un système à base de connaissances. C'est pourquoi, dans le cadre de notre étude, il nous a paru intéressant de nous inspirer des études menées dans le domaine de l'ingénierie des connaissances, sans toutefois en respecter scrupuleusement la terminologie. Dans ce contexte, les travaux de Newell [Newell 1982] pour le développement de systèmes experts de seconde génération constituent une référence incontournable. Ces travaux sont détaillés et critiqués dans [Pierret 1996] par exemple. Dans le cadre du développement de systèmes à base de connaissances, ils préconisent de

faire une distinction nette entre 2 « niveaux » : le « **niveau connaissance** » et le « **niveau symbole** » [Newell 1982] :

*« Le niveau connaissance est un niveau abstrait qui traite du contenu de la connaissance. A ce niveau, il s'agit de discuter du contenu de la connaissance qui doit être exprimée, et non de la manière dont la connaissance doit être codée dans un langage de représentation ».*

*« Le niveau symbole traite de la manipulation des symboles représentant la connaissance. Au niveau symbole, il s'agit de définir les structures de données informatiques capables de coder le contenu défini au niveau connaissance, connaissances qui sont exprimées dans l'ensemble des concepts définis au niveau symbole ».*

Selon Newell, la conception d'un système à base de connaissances passe par une décorrélation de ces deux niveaux. Dans cette partie de la thèse, nous suivrons ces principes en nous plaçant successivement à ces deux niveaux. Cette seconde partie s'articulera autour de quatre chapitres :

- Dans un premier temps, nous nous placerons au « niveau connaissance » en tentant d'identifier l'ensemble des connaissances qui sont mises en jeu dans un processus d'interprétation de documents techniques. Nous décrirons ces connaissances suivant trois grandes catégories : les connaissances descriptives qui concernent les données manipulées (« Quelles sont les données présentes sur le document ? Comment sont-elles agencées ? »), les connaissances opératoires qui concernent les traitements (« Quel est le traitement adéquat pour cette situation ? Comment en régler les paramètres ? ») et les connaissances stratégiques, qui concernent l'agencement des traitements (« Quel est le but ? Quels traitements faut-il enchaîner pour y parvenir ? »). Il est important de noter que ceci est une catégorisation parmi d'autres. Nous verrons par la suite que dans chacune de ces « catégories », il est possible de trouver une nouvelle décomposition, en fonction des compétences mises en jeu.
- Après la présentation de cette liste catégorisée, le second chapitre de cette partie revêt un caractère bibliographique. En nous appuyant sur les connaissances identifiées dans la première partie, nous proposerons une synthèse de l'état de l'art en terme de représentation de ces connaissances dans les systèmes d'interprétation de documents techniques. Puis, après un bilan concernant ces systèmes, nous analyserons les approches adoptées par quelques systèmes de vision, dont la problématique est proche de la nôtre. Dans ce cadre, une vision globale des formalismes adoptés

dans la littérature correspondante sera d'abord présentée, avant une description détaillée de trois systèmes qui ont inspiré nos travaux.

- Le troisième chapitre de cette partie concerne les premières propositions et implémentations effectuées dans le cadre d'un premier système *NATALI v1*, développé dans notre équipe depuis 1998. Celui-ci sera présenté en adoptant la même structuration que celle utilisée pour l'état de l'art. Nous en dresserons une analyse critique, basée sur l'analyse de la littérature et sur de premiers cas d'usage. Cela mettra en avant les problématiques principales liées à *NATALI v2*.
- Les choix correspondant à cette problématique seront alors présentés dans le quatrième chapitre. De premières propositions d'un point de vue acquisition et représentation de connaissances seront exposées. Nous montrerons la plus-value apportée par une approche à base de connaissance du point de vue interaction système/environnement. Nous proposerons également dans cette partie de nombreux axes de poursuite de ces travaux en terme de « connaissances ».

Une conclusion sur l'aspect ingénierie des connaissances dans les systèmes d'interprétation permettra de clore cette seconde partie du manuscrit. Nous y analyserons nos contributions, en les situant par rapport à la littérature du domaine.

## **3.2. Les connaissances d'un système d'interprétation**

### **3.2.1. Introduction**

Dans ce premier chapitre, nous nous plaçons au « niveau connaissance » de Newell. Nous cherchons donc à dresser une liste exhaustive et catégorisée des connaissances mises en jeu dans un processus d'interprétation, sans évoquer la manière de coder celles-ci dans un langage de représentation. Pour ce faire, nous nous appuyons sur des simulations d'un fonctionnement idéal du système, en listant les connaissances que celui-ci utilise alors pour avoir le comportement souhaité. Cette réflexion nous a poussé à catégoriser les connaissances d'un système d'interprétation en 3 principaux types : descriptives, opératoires, stratégiques.

### **3.2.2. Connaissances sur les données (connaissances descriptives)**

Les connaissances « descriptives » ont pour but de fournir un modèle du contenu du document qui doit être interprété. C'est l'ensemble des connaissances symboliques et numériques décrivant les différentes entités manipulées par le système (les objets sémantiques, leurs caractéristiques, les relations les liant, les primitives images ...). Pour identifier ces connaissances, il nous a paru opportun de revenir sur l'essence même des documents techniques, à savoir le fait qu'ils constituent des vecteurs d'échange entre les différents protagonistes qui les manipulent. Les différents intervenants dans le cycle de vie du document [Lorette 1992][Labiche 1998] (Figure 3.1) ayant chacun des expertises diverses, il est possible de catégoriser les connaissances descriptives en fonction des compétences mises en jeu au cours de ce cycle. En phase de production, le réseau a d'abord été pensé ; son organisation respecte des règles précises en terme métiers (fonctionnalités, agencement... ). Le premier acteur de la production du document est donc le spécialiste du domaine qui a défini ces règles. Les connaissances de celui-ci doivent être utilisées dans le processus de consommation du document. Le second acteur principal du processus de production est le dessinateur. Celui-ci doit généralement respecter une normalisation de représentation, afin que le document puisse être appréhendé par son ou ses lecteurs. Cette normalisation doit être incluse dans le processus de consommation, afin que le consommateur puisse associer les éléments graphiques à des objets du domaine. Les connaissances descriptives peuvent donc être décomposées en connaissances métier et en connaissances graphiques, ce qui mène à une première scission du modèle. Bien sûr, cette décomposition peut encore être affinée si les compétences citées ci-dessus peuvent être réparties sur différentes spécialités (juridiques, techniques...). Néanmoins, cette scission métier / graphique permet de rester générique, puisque qu'elle est valable dans toute production de documents techniques.



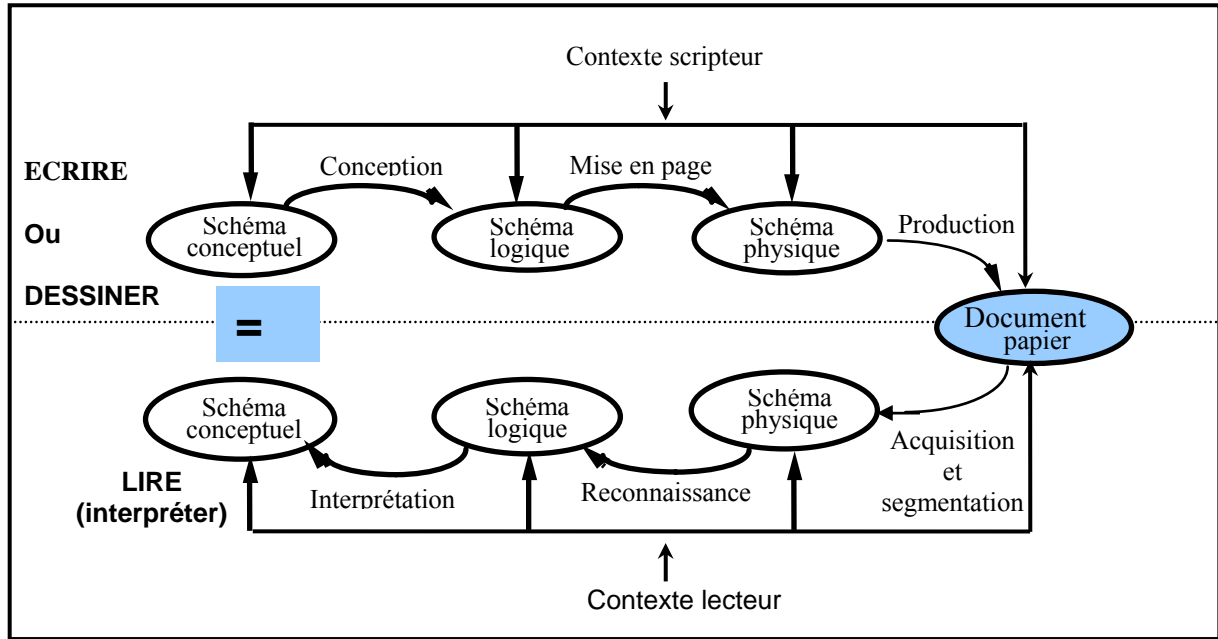


Figure 3.1 : Le cycle de vie d'un document suivant [Lorette 1992] [Labiche 1998]

### 3.2.2.1. Les connaissances descriptives « métier »

Pour interpréter correctement un document aussi riche en informations qu'un plan d'itinéraire, connaître la relation entité / représentation graphique n'est pas suffisant. En effet, les infrastructures qui y sont représentées respectent des règles précises quant au métier auquel elles font référence, en terme de fonctionnalité par exemple. Ne pas exploiter ce type d'information ne rendra pas le système aussi « intelligent » qu'il peut l'être. C'est pourquoi, dans cette partie, nous essayons de dresser une liste des connaissances descriptives métier mises en jeu, dans le cas particulier de France Telecom, afin de les intégrer à notre système. Certaines de ces connaissances peuvent être généralisées à tout type de documents techniques. Le cœur de ces connaissances réside en effet dans les objets du domaine, leurs propriétés, et leurs interactions. C'est une catégorie propre à une application donnée, qui nécessite donc l'expertise d'un spécialiste du domaine. Cela concerne par exemple :

- Les connaissances sur le document traité, que représente-t-il ? Quelle est son utilité, son rôle dans le Système d'Information (S.I.) de l'entreprise ? Quel est son cycle de vie [Labiche 1998] ?
- Les connaissances sur les objets du domaine existants (sous répartiteur, point de concentration, tronçons d'artères, chambre dans le cas d'un document France Telecom). Un système d'interprétation doit connaître l'ensemble des données qu'il pourra détecter dans un plan.
- Les connaissances sur les propriétés de ces objets au niveau métier (un point de concentration est décrit par son nombre de paires, un tronçon d'artères est décrit par sa longueur dans la rue...).
- Les connaissances sur le rôle de ces objets (une chambre contient des réglettes permettant d'interconnecter les câbles de différents tronçons d'artère...).

- Les connaissances sur les relations (sémantiques, spatiales sur le terrain...) entre les objets métier. Ces connaissances intègrent également l'interaction entre différents opérateurs de réseau (Electricité De France, Gaz De France...). Ces derniers peuvent en effet, par l'intermédiaire de plans, apporter des connaissances pour l'interprétation.
- Les connaissances sur les parties du SI de l'organisation et sur la présence d'éventuels objets intéressants ou d'attributs pour ces objets dans des bases de données existantes.

Les exemples cités ci-dessus montrent que dans cette catégorie n'entrent en ligne de compte que les informations de type « métier », issues de l'expertise du spécialiste du domaine. Dans le cadre du développement d'un système de rétro conversion, cette expertise devra être saisie par ce spécialiste qui alimentera un formalisme de représentation simple, grâce à une IHM conviviale. Cette saisie devra dans un premier temps selon nous, être décorrélée des représentations graphiques associées, les connaissances étant de natures différentes. Les connaissances issues de cette saisie permettront de valider le résultat obtenu par l'interprétation. A titre d'illustration, une chambre (élément d'infrastructure du réseau France Telecom permettant la répartition des câbles) reconnue dans une région sans habitations d'après des connaissances issues du fond de plan ne pourra être validée. Ces connaissances du domaine sont donc fondamentales au processus d'interprétation. Néanmoins, elles ne sont pas suffisantes. En effet, le système ne pourra interpréter un document que s'il a des connaissances sur cette partie graphique associée à chacune des entités précitées.

### **3.2.2.2. Les connaissances descriptives graphiques**

Au même titre qu'un humain, un système n'est pas capable de reconnaître un objet sans connaissances *a priori* sur sa représentation graphique [Varela 1989]. Pour l'interprétation de l'image scannée d'un document, il est donc nécessaire de connaître la représentation graphique des entités du domaine concerné. En effet, dans le contexte de l'interprétation, celles-ci sont représentées par leur image numérisée. Cette image peut être disponible sous différents codages (*bitmap* en niveaux de gris, binaire ou couleur, *run length* ...) et avec des résolutions qui peuvent être variables. En outre, si cette représentation sous forme d'image peut parfois être directement utilisée, la reconnaissance des objets passe dans la plupart des cas par d'autres types de représentation en dérivant (cf. Partie 2) : une liste de points (ses contours, son squelette), un graphe de vecteurs (issus du squelette ou des contours), un graphe d'occlusions, un vecteur statistique de caractéristiques... Selon la vue adoptée, le modèle d'une entité peut alors être une image modèle, un réseau de neurones ayant subi un apprentissage, un graphe modèle... Il est aussi important de considérer le fait que certaines représentations soient adaptées à certaines données et pas à d'autres. A titre d'illustration, les câbles sur les documents France Telecom seront plutôt décrits structurellement alors qu'un caractère sera plus souvent représenté statistiquement. La connaissance sur la représentation des entités est donc fortement liée aux stratégies et traitements d'images. Cette connaissance doit être acquise par coopérations entre un expert du

domaine, qui montre l'image des entités, et l'expert de traitement des images qui choisira une ou plusieurs représentations appropriées en fonction des stratégies de traitement qu'il jugera efficace pour l'extraction. En effet, l'utilisateur du système ne pourra pas définir les éléments à rechercher au moyen d'un vecteur de caractéristiques statistiques. Le problème de l'acquisition de connaissances doit donc absolument être pris en compte pour le choix de la méthode de représentation d'un objet. De plus, outre la représentation graphique des entités, il est aussi nécessaire, pour le système, d'avoir des connaissances sur les relations spatiales graphiques entre les différents objets (orientation commune, connexion possible, impossible...).

### **3.2.2.3. Bilan concernant les connaissances descriptives**

Les deux paragraphes précédents illustrent le fait que les connaissances descriptives doivent inclure la description au niveau métier des entités qui vont être manipulées, mais aussi la représentation graphique de ces entités. Au niveau représentation, il sera donc nécessaire de prévoir des structures de données permettant au système de recueillir l'expertise métier de la documentation, mais aussi différents modèles graphiques s'adaptant aux différents types de données. En phase d'acquisition, l'expert d'un domaine donné ne peut pas définir un vecteur de caractéristiques modèle ou un graphe modèle, puisqu'il ne dispose pas des compétences de traitement d'image qu'il faudrait pour cela. Cette partie est pourtant importante puisque c'est sur cette description que le système va baser la stratégie permettant de répondre à une requête utilisateur. Cela montre bien l'importance de la prise en compte de la phase d'acquisition des connaissances dans le choix de la (ou des) méthode(s) de représentation des connaissances. Notons aussi que certaines des connaissances descriptives graphiques peuvent se déduire des connaissances métier mais que ce n'est pas toujours le cas. En effet, savoir qu'un PC doit être proche d'une chambre d'après le modèle métier ne donne pas d'indication sur le fait qu'il soit ou non connecté sur le dessin.

### **3.2.3. Connaissances sur les traitements (connaissances opératoires)**

Pour appliquer une stratégie de traitement en fonction d'une requête initiale, un système d'interprétation doit disposer d'une boîte à outils de traitement d'images mais aussi d'un certain nombre de connaissances sur les outils qu'elle contient. Le but de cette partie est d'en dresser une liste en réutilisant la catégorisation précédente graphique / métier.

#### **3.2.3.1. Au niveau « métier »**

Au niveau du domaine, les connaissances opératoires concernent les méthodes d'associations d'entités, en fonction de leur relation métier. Elles consistent à décrire l'efficacité des méthodes d'associations d'objets en fonction des relations exprimées dans les connaissances descriptives du domaine.

### **3.2.3.2. Au niveau traitement d'image**

Au niveau du traitement d'image, un système d'interprétation idéal doit selon nous intégrer :

- Des connaissances sur les entrées, les sorties et les paramètres des algorithmes
- Des connaissances sur la loi de réglage des paramètres en fonction d'un contexte, de contraintes et/ou de son historique, ce qui exige donc :
- Des connaissances sur les contextes possibles, et leur caractérisation
- Des connaissances sur les contraintes applicables à un algorithme (temps, qualité, spatiales ...)
- Des connaissances sur l'histoire des exécutions de l'algorithme (par l'intermédiaire de retours d'évaluation obtenus) en fonction des contextes et des contraintes.
- Des connaissances sur l'évaluation de la qualité d'un résultat de traitement

### **3.2.4. Connaissances sur les stratégies (connaissances stratégiques)**

Les traitements intégrés et modélisés dans le système doivent, si le but est de concevoir un système dynamiquement adaptable, être utilisés de façons différentes en fonction du contexte de traitement en cours, et de l'application considérée. L'aspect stratégie d'interprétation, c'est à dire le choix de la séquence de tâches à effectuer, revêt donc un aspect fondamental pour parvenir à satisfaire un but d'interprétation sous des contraintes données. Il est nécessaire, pour que le système soit à la fois compréhensible et adaptable, de modéliser les différents types de stratégies utilisables. Nous listons ci-dessous un certain nombre de connaissances du système concernant les stratégies.

- Des connaissances sur les tâches possibles du système. Elles regroupent l'ensemble des stratégies possibles d'utilisation du système. En effet, en fonction des attentes de l'utilisateur, le but du système peut être de rechercher un objet donné dans l'image du document, d'en extraire des caractéristiques, de tout interpréter...
- Des connaissances sur les contraintes possibles à appliquer à chaque tâche. L'utilisateur du système doit pouvoir imposer une recherche dans une zone donnée, en respectant un temps de recherche...
- Des connaissances au niveau métier des méthodes d'évaluation du contexte pertinent à évaluer pour la construction d'un plan.
- Des connaissances sur la méthode de traduction d'une tâche en sous tâches. Elles consistent à savoir lire le modèle de connaissances descriptives pour déterminer les données à extraire en priorité, en tenant compte des liens de type Est\_Constitué\_De ou Est\_Associé\_A par exemple et en utilisant le contexte des données d'entrée.
- Des connaissances sur la planification (ordonnancement) de tâches : possibilité de compétitions puis de fusions, de collaborations... Si une tâche est divisée en deux sous tâches, la première doit-

elle être effectuée avant la seconde sans collaboration ? Les deux réalisées en coopération ? Ou bien l'une puis l'autre avec indices extraits de la première pour appliquer des contraintes ?

- Des connaissances sur le rôle de l'utilisateur : quand et comment son intervention peut-elle être sollicitée ?
- Des connaissances sur l'acquisition de connaissances, pour prendre en compte de nouveaux objets, de nouvelles méthodes.
- Des connaissances sur les procédés d'extraction d'informations provenant d'autres éléments du SI

### **3.2.5. Conclusion**

Dans ce chapitre, nous avons dressé une liste catégorisée, la plus exhaustive possible, des différents types de connaissances qui peuvent être mis en jeu au cours d'un processus d'interprétation de documents techniques. Nous avons relevé 3 principaux types de connaissances :

- Les connaissances descriptives, qui concernent les données traitées par le système.
- Les connaissances opératoires, qui concernent les traitements manipulant ces données.
- Les connaissances stratégiques, qui prennent en considération l'agencement de ces traitements et les collaborations avec l'utilisateur, en vue de satisfaire un but.

Il est bien évident que ces différentes catégories ne peuvent être complètement séparées en terme de représentation. Le choix effectué pour modéliser les traitements sera tributaire du choix en ce qui concerne les données puisque les traitements vont manipuler celles-ci. Cette dualité traitement / données, mais aussi l'interaction avec l'aspect stratégique rend très complexe la représentation des connaissances. Ce point est souvent soulevé dans les travaux sur l'acquisition et la représentation des connaissances pour la vision [Baujard 1992], [Ovalle 1991]. Dans le chapitre qui suit, nous verrons quels ont été les choix des auteurs pour la représentation des connaissances, pour les systèmes d'interprétation de documents techniques de la littérature, ainsi que pour quelques systèmes de vision.

### **3.3. Les connaissances dans la littérature du domaine**

Ce chapitre a pour but de dresser un état de l'art concernant la problématique de la modélisation et de la représentation des connaissances dans les systèmes d'interprétation. Dans une première section, nous nous consacrons aux systèmes d'interprétation de documents techniques. Néanmoins, les systèmes dédiés à de tels documents ayant pris en compte l'aspect de représentation explicite de la connaissance sont rares dans la littérature qui s'attache davantage aux outils de traitement d'images ou d'extraction d'indices. C'est pourquoi une seconde section est dédiée à l'analyse plus large des systèmes de compréhension d'images (scènes naturelles, biomédicales, scènes d'aéroport...). En effet, ce domaine, bien que différant du nôtre sur divers aspects, n'en reste pas moins proche en terme de modélisation d'objets, de traitements et de stratégies.

#### **3.3.1. Dans les systèmes d'interprétation de documents techniques**

Cette première section décrit huit systèmes d'interprétation de documents techniques de la littérature. Cet état de l'art n'est évidemment pas représentatif de tous les travaux menés dans le domaine de l'analyse de documents techniques, tant le nombre de travaux portant sur les opérateurs est important. Toutefois, il prend en considération les systèmes pour lesquels un effort a été consenti sur l'aspect explicitation des connaissances manipulées. Etant donnée la connexité de ces travaux aux nôtres, nous avons fait le choix de détailler successivement chacun des systèmes, en prenant le point de vue ingénierie des connaissances. Une synthèse est ensuite proposée dans le dernier paragraphe.

##### **3.3.1.1. Le système de Den Hartog [Den Hartog 1996]**

C'est un système développé dans les années 90 ayant pour but l'interprétation du cadastre néerlandais. C'est, à notre connaissance, l'un des premiers systèmes d'interprétation de documents techniques à avoir mis l'accent sur l'aspect explicitation des connaissances, dans le but de réduire le temps d'adaptation du système à une autre application. Dans ce système, un unique formalisme est utilisé pour représenter la connaissance *a priori*. Il s'agit des réseaux sémantiques, qui avaient déjà été utilisés dans les premiers travaux sur la vision [Winston 1970]. Dans [Den Hartog 1996], ils permettent la représentation de deux des catégories de connaissances citées dans la partie précédente :

- Les connaissances descriptives

Un réseau sémantique est un graphe formé de nœuds, représentant des concepts, et d'arcs orientés décrivant les relations sémantiques unissant ces concepts. Pour l'équipe de Den Hartog, chaque nœud dénote la description géométrique d'un type d'objet susceptible d'apparaître dans l'image du document. Les arcs décrivent quant à eux les relations spatiales entre ces types d'objets. Un exemple de réseau ainsi construit est présenté Figure 3.2. Dans ces réseaux, les types d'objets sont décrits par

un nom, par un ensemble de propriétés (des informations géométriques, essentiellement déduites du rectangle englobant), et par un intervalle de valeurs plausibles pour chacune de ces propriétés.

Les liens dans le réseau, (*i.e.* les relations spatiales entre objets) sont munis des propriétés suivantes : le type de la relation (essentielle, optionnelle ou impossible), les identifiants des deux objets liés par la relation, la taille de l'espace de proximité possible, l'angle entre les deux objets, (défini à partir du rectangle englobant), et deux drapeaux témoignant des possibilités d'inclusion et de chevauchement. Une priorité est également accordée à chacune des relations dans le réseau, mais celle-ci est plutôt une connaissance stratégique puisqu'elle permet l'ordonnancement des traitements.

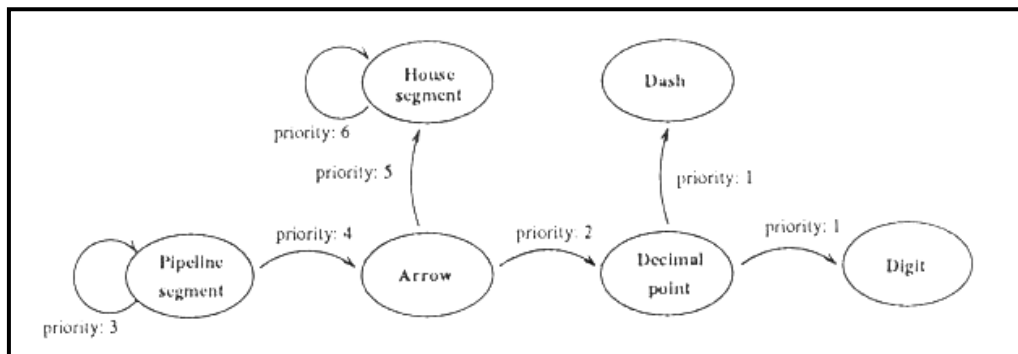


Figure 3.2 : Le réseau sémantique de Den Hartog

- Les connaissances stratégiques

Certaines des connaissances stratégiques évoquées en 3.2.4 sont explicitées dans le réseau sémantique de Den Hartog, sous la forme de priorités de voisinage associées aux objets. Un mécanisme basé sur ces relations permet alors de générer une liste ordonnée par priorité des objets à rechercher à la suite de chaque nouvelle reconnaissance d'objet. La séquence de traitement à effectuer est donc automatiquement générée. De plus, le réseau est également utilisé pour la gestion de la cohérence des résultats d'interprétation. A partir de la propriété « type de relation » de chacun des liens, un objet sera jugé incohérent s'il est voisin d'un objet alors que la relation unissant les deux types est « impossible ». De même s'il n'est pas voisin d'un objet alors que leur relation de voisinage est « essentielle ». Dans un tel cas de figure, l'intervention de l'opérateur sera sollicitée. Aucun mécanisme automatique de correction des incohérences ne semble en effet intégré dans le système.

En guise de bilan, ce système est probablement l'un des plus avancés dans le domaine de l'interprétation de documents techniques. Cet état de fait est sans conteste dû aux avancées réalisées dans l'explicitation des connaissances. Le formalisme adopté pour représenter la connaissance *a priori* du système (le réseau sémantique) témoigne de son efficacité pour représenter de façon compacte à la fois des connaissances descriptives et des connaissances stratégiques. Ceci permet, selon l'auteur, une réduction considérable du temps d'adaptation du système à une nouvelle application. Néanmoins, l'auteur se reproche de ne pas avoir suffisamment intégré dans son modèle les notions de hiérarchie,

de spécialisation et de décomposition entre les objets modélisés. A titre d'exemple, un concept de type « cote », concrétisé sur un schéma par un toponyme associé à une flèche ne peut être intégré dans le système en tant qu'objet. De plus, entre deux objets du réseau, seules les relations spatiales sont prises en compte, en oubliant les autres liens qu'il peut y avoir entre deux objets (la longueur d'un pipeline sur le dessin et sa cote par exemple). Notons également que l'aspect opératoire de la connaissance a été mis de côté dans ce système, l'auteur n'abordant pas la modélisation des traitements, le réglage de leurs paramètres...

### **3.3.1.2. Le système de Joseph [Joseph 1992]**

Ce système, nommé ANON, est un système d'analyse de documents à base de connaissances qui a pour but d'extraire des éléments graphiques 2D dans une image en niveaux de gris de dessin d'ingénierie. L'approche adoptée est basée sur la combinaison de *schemas* décrivant les prototypes des objets avec une librairie d'opérateurs de traitements d'images et un *parser* basé sur une grammaire à précedence faible. Ce système intègre dans ce cadre des stratégies de traitements mixtes (ascendantes et descendantes) dans une structure simple et flexible, inspirée du modèle de cycles perceptifs humains. Dans l'article décrivant ce système, Joseph met en évidence le fait que pour aboutir à un système « intelligent » d'interprétation de documents d'ingénierie, une modélisation explicite du contenu du document est fondamentale. De ce point de vue, il pense, comme notre équipe, que les systèmes de compréhension de scènes peuvent apporter des idées et des méthodes aux recherches concernant les documents. C'est pourquoi Joseph s'inspire, pour la conception de son système, de travaux tels que SIGMA [Matsuyama 1985], MAPSEE [Mulder 1987], ACRONYM [Brooks 1981, 1983] ou GOLDIE [Kohl 1987], qui sont des systèmes que nous évoquerons dans la section suivante.

- Les connaissances descriptives

Parmi les idées inspirées de ces systèmes, citons la modélisation des objets par des classes de *schemas* qui constituent le cœur du système (ce terme de *schema*, que nous retrouverons plusieurs fois est issu du système VISIONS [Hanson 1978]). Ces structures de données, très proches des *frames* de Minsky [Minsky 1975], permettent la description des prototypes d'objets à extraire. Les classes définies au moment de la parution de l'article correspondent aux lignes (continues, chaînées ou pointillées), aux courbes, aux hachures, au texte (lettres et mots), et à des entités de plus haut niveau telles que les cotations... Comme dans tout mécanisme objet, lorsqu'une telle entité est détectée par le système, celle-ci est représentée par une instance de la classe correspondante. Dans chacun des *schemas* définis, une description de la construction de l'objet, un ensemble de variables d'état, et un certain nombre de procédures et fonctions écrites en C sont présents. Ces dernières peuvent être scindées en deux catégories. Les premières sont des fonctions dites administratives et permettent la construction d'un objet, l'accès à ses différents attributs et la modification de ses variables d'état. La seconde partie des fonctions permet, quant à elle, l'interface avec les opérations de traitement d'images. Les différents



*schemas* définis ne sont, bien entendu, pas isolés, mais forment un réseau dans lequel les arcs représentent des relations de décomposition (Figure 3.3) et de spécialisation (Figure 3.4). La relation de composition permet une représentation structurale en plus de la description géométrique des objets.

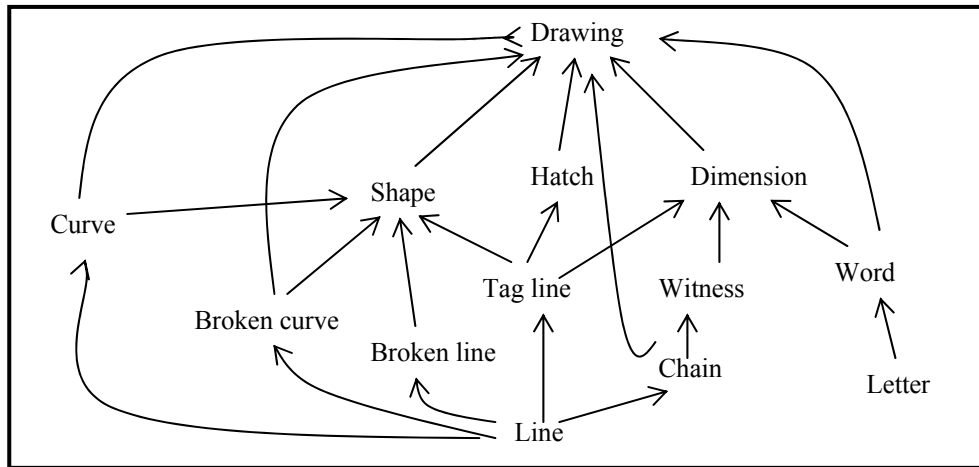


Figure 3.3 : Les relations de sous partie entre les schémas du réseau

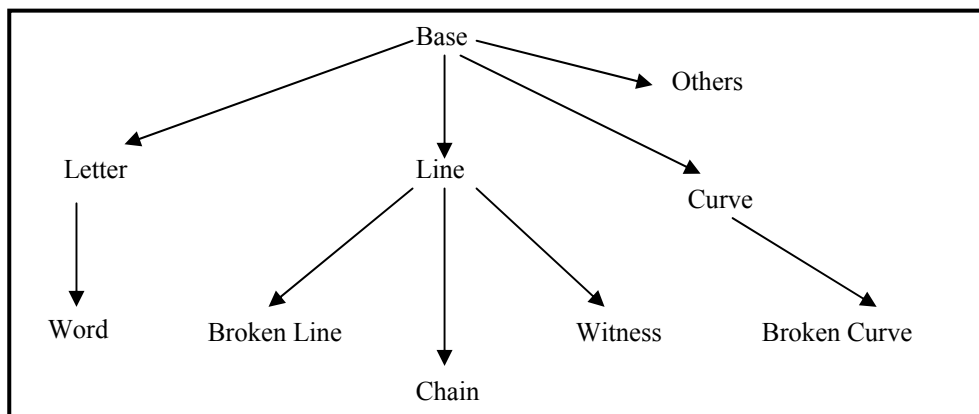


Figure 3.4 : Les relations d'héritage dans ANON

- Les connaissances opératoires

Dans [Joseph 1992], les traitements sont intégrés sous forme de fonctions membres dans les *schemas*, faisant ainsi le lien entre les objets et les routines d'une librairie. Deux types de fonctions sont distingués : celles qui recherchent des germes, en utilisant des recherches de primitives telles que segments et arcs, et celles qui utilisent ces germes pour poursuivre la construction des objets. La notion de contexte d'utilisation des traitements et du réglage des paramètres en fonction de ce contexte est absente de ce système. Concernant les traitements, notons que pour la recherche de germes, l'image en niveaux de gris est directement utilisée, sans seuillage ni squelettisation préalable, ce qui constitue une originalité de ce système.

- Les connaissances stratégiques

Dans [Joseph 1992], 191 règles écrites sous la forme d'une grammaire à précedence faible sont utilisées pour gérer l'enchaînement des instanciations de *schemas*. Cette grammaire est traitée par un

analyseur syntaxique issu de l'utilitaire UNIX *yacc* (**y**et **a**nother **c**ompiler **c**ompiler). Cette grammaire ne contribue pas à la définition des objets graphiques, mais plutôt à la stratégie de détection de ceux-ci. Les règles définissent les séquences d'événements acceptables, ces événements étant le flux de *tokens* (objets instanciés) générés au fur et à mesure de l'interprétation. En terme d'acquisition, l'utilisation de *yacc* permet un prototypage rapide des stratégies, grâce à son interface utilisateur flexible et à son mécanisme de vérification de la cohérence des règles définies. Parmi les avantages d'une telle approche, [Rao 1988] cite la simplicité, le format expressif, la précision formelle, et la garantie de cohérence. Néanmoins, la définition des règles ne semble pas triviale et l'auteur juge lui-même que son système a des lacunes en terme de généralité. De plus, il n'y a pas dans ce système de méta-règles gérant le déclenchement des règles.

### **3.3.1.3. Le système de Pasternak [Pasternak 1993]**

Le système de Pasternak aborde lui aussi le problème de la conversion de dessins papier en représentations utilisables par des applications informatiques. Alors que les approches précédentes ont été capables de résoudre ce problème dans des domaines d'application spécifiques, le noyau du système présenté dans ce papier se veut être adaptable à une large variété d'applications, incluant les dessins d'ingénierie et les schémas électriques. L'élément clef de l'approche adoptée est un processus d'interprétation à base de connaissances, exploitant un langage de spécification pour définir les objets graphiques cohérents à reconnaître sur les dessins. Les problèmes de complexité éventuellement engendrés par une approche strictement déclarative sont résolus par une compilation automatique des spécifications graphiques en procédures efficaces. L'adaptabilité du système noyau est montrée en utilisant des exemples de domaines différents. En terme de connaissances, l'auteur pense qu'une approche orientée objet offre le meilleur potentiel pour un système d'interprétation multi-domaines et pour éviter les déficiences d'un codage procédural de la connaissance géométrique. Pour cela, ADIK fournit un ensemble de primitives géométriques pré-spécifiées et d'éléments de spécification. La base de connaissances contient la description des objets graphiques et les règles géométriques du domaine basées sur ces éléments. Cela permet de construire des objets génériques et de les associer dans une hiérarchie de décomposition. Puis, l'ensemble des spécifications géométriques de chaque objet dans la base de connaissances est compilé en un ensemble de fonctions de contraintes, qui sont attachées aux objets génériques. Finalement, la taxonomie et les relations de décomposition entre les différents objets sont exploitées pour déterminer la stratégie de contrôle de l'interprétation. Nous détaillons ces différents points ci-après.

- Les connaissances descriptives

Trois types d'objets sont exploités dans une première couche : les lignes, les arcs et les chaînes de caractères. Pour chacun de ces types, trois primitives géométriques sont associées (position, orientation et étendue), elles permettent à l'utilisateur du domaine considéré de définir les objets à

rechercher, en utilisant un langage de spécification purement déclaratif. Ce langage, dont l'auteur précise qu'il est plus simple d'utilisation qu'un langage de programmation, repose sur ces primitives. Elles sont associées grâce à un ensemble de relations et de prédicats, à un mécanisme de tolérance, et à des éléments structurels qui décrivent les relations de composition et de spécialisation. Les relations de dépendance géométrique entre objets sont définies en terme de relation d'attributs entre primitives géométriques grâce aux prédicats ISPOS (position), ISORI (orientation), ISARC (angle), ISLEN (distance), LINE (ligne), LOC (position relative) et IDENT (objets identiques). A partir de ce langage de spécification, les connaissances sont alors compilées en un ensemble de procédures exécutables.

- Les connaissances opératoires

Les connaissances opératoires de ce système ne sont pas explicitées clairement, du fait de la génération de code exécutable à partir de la spécification des objets. Les traitements et leur paramétrage en fonction d'un contexte donné ne semblent pas avoir donné lieu ni à une modélisation, ni à une représentation.

- Les connaissances stratégiques

Le processus d'interprétation dans ADIK commence par la lecture de l'ensemble des primitives issues du premier étage de traitement. En fonction des éléments disponibles, un mécanisme de trigger est utilisé pour générer la séquence des traitements appliqués. Un objet du domaine sera instancié uniquement si tous ses composants ont été détectés. Il s'agit donc d'un exemple de processus dans lequel il n'y a pas de stratégie prédéfinie, le contrôle du système est opportuniste. Il ne semble pas y avoir de notion explicite d'intention ou de génération d'hypothèses (approche descendante) dans ce système, l'objectif est toujours une interprétation complète du document.

En guise de bilan, le point fort de ce système est selon nous la représentation explicite des connaissances graphiques sur les objets à extraire des documents. Cela va totalement dans le sens de la genericité d'un système puisqu'un utilisateur peut lui-même définir les entités à extraire. Notons que cette approche est à la base du système décrit en 3.3.1.5.

#### **3.3.1.4. Le système de Yu [Yu 1997]**

Ce système, développé par l'équipe de Yuhong Yu pour *Lucent Technologies*, vise à reconnaître une large gamme de schémas d'ingénierie tels que les diagrammes de circuits logiques, les schémas électriques... Le postulat de fonctionnement de ce système repose sur la possibilité de décomposer le contenu du document à reconnaître en un ensemble de symboles interconnectés par des lignes. En terme de connaissances, seules les connaissances descriptives sont modélisées dans ce système. En effet, ni les outils de traitement d'images, ni la stratégie (purement ascendante) ne sont explicités dans ce système.

- Les connaissances descriptives

Dans ce système, les symboles du domaine sont représentés par une décomposition structurelle hiérarchique suivant 3 niveaux. Le premier niveau contient les symboles. Sur un second niveau, chacun des symboles du premier niveau est décomposé en l'ensemble des formes connexes le constituant. Puis, chacun de ces éléments constitutifs est représenté par deux vecteurs de caractéristiques, basés sur leur décomposition en deux types de primitives : les traits et les occlusions (entités de bas niveau). Les traits et occlusions modèles sont stockés sous forme d'images normalisées de 64\*64 pixels dans les différentes orientations pouvant apparaître sur l'image du document. Le modèle (vecteur de caractéristiques) de chacun des « éléments constitutifs » des symboles est alors un simple compteur du nombre d'occurrences de chacun des modèles de primitives de base dans le dessin. Trois propriétés sont ajoutées à ce vecteur : le plus grand élément de base, sa taille, et le nombre de lignes de connexion associées à l'élément.

En ce qui concerne les connaissances opératoires et stratégiques, rien n'est explicité dans ce système. Les opérateurs de traitement ne sont pas modélisés et la stratégie de traitement est totalement figée et ascendante. Toute la connaissance liée à ces points est donc représentée directement dans le code, ce qui nuit probablement à la généralité de ce système.

### **3.3.1.5. Le système de Ah Soon et Dosch [Ah Soon 1998][Dosch 2000]**

Ce système, développé par Christian Ah Soon et Philippe Dosch sous la direction de Karl Tombre dans le cadre du projet I.S.A. (Image, Synthèse et Analyse) au LORIA (Laboratoire LOrrain de Recherches en Informatique et ses Applications) s'intéresse à l'analyse de plans architecturaux. En terme de connaissances, d'importants efforts ont été produits concernant la modélisation explicite de la symbologie présente sur ce type de documents. Cette approche permet d'obtenir un système souple, générique et facilitant l'acquisition de la description des symboles à extraire. Deux modèles différents sont en fait utilisés dans ce système. Le premier, visant à rendre plus simple et plus conviviale l'acquisition, est basé sur un langage de description inspiré des travaux de Pasternak (cf. 3.3.1.3.). Une grammaire, plus simple que celle de ce dernier, a été définie (Figure 3.5). Elle permet de décrire les symboles à rechercher, grâce à un ensemble de contraintes que doivent vérifier les segments et arcs. Le second modèle est dynamiquement généré à partir du premier. Il a pour but d'optimiser la recherche des symboles décrits dans l'image vectorisée en phase de reconstruction. Il s'agit d'un réseau de contraintes, inspiré des travaux de Messmer [Messmer 1993][Messmer 1996], qui a pour principal point fort de factoriser les contraintes communes aux différents symboles. Cette factorisation permet alors d'accélérer la recherche d'isomorphismes de sous graphes pour la détection et la

reconnaissance des symboles. En effet, plus le nombre de nœuds communs est important, plus la recherche dans le graphe est réalisée rapidement.

```

symbole ↦ # type SEG entier ARC entier {contrainte;}+ instantiation
contrainte ↦ contrainte ou contrainte | contrainte et contrainte
contrainte ↦ point == point | réel op_comp réel
réel ↦ réel | réel cm | min ( réel , réel ) | max ( réel , réel )
réel ↦ abs ( réel ) | sin ( réel ) | cos ( réel ) | tan ( réel )
réel ↦ point.x () | point.y () | point.distance ( point )
réel ↦ prim.f_elem () | prim.f_angle( prim ) | réel op_alg réel
réel ↦ arc.rayon () | point.angle ( point, point )
f_elem ↦ longueur | epaisseur | x | y
f_angle ↦ pt1_angle_pt1 | pt1_angle_pt2 | pt2_angle_pt1 | pt2_angle_pt2
point ↦ prim.point1 () | prim.point2 () | arc.centre ()
prim ↦ arc | segment
arc ↦ arc entier
segment ↦ seg entier
    
```

Figure 3.5 : La grammaire de définition des symboles dans Ah Soon [Ah Soon 1998]

Une modélisation à base de textures des symboles est également présentée dans le cadre de ce système [Dosch 2000]. Le principe consiste d'abord à extraire des polygones à partir de la vectorisation puis à construire progressivement un graphe. Dans ce graphe, les nœuds sont des ensembles de polygones proches au sens d'un critère de surface et les arcs sont les relations d'adjacence entre ces ensembles de polygones. Ces graphes sont stockés dans une pyramide irrégulière permettant de représenter la hiérarchie des regroupements effectués, et finalement les textures détectées. A partir de cette reconnaissance de texture, les symboles sont représentés par les bornes *min.* et *max.* du nombre de polygones de la texture représentant l'objet.

Au niveau du domaine, des modèles 2D (pour les étages) et 3D (pour les bâtiments) sont utilisés. Le modèle 2D (modèle d'un étage) est basé sur les résultats de la reconnaissance 2D et sur la hauteur de ces derniers (connue *a priori*). Sept familles d'objets ont été identifiées (fenêtre, fenêtre française, portes, murs porteurs, murs de séparation, escaliers et conduites). Parmi ces objets, l'auteur a cherché à déterminer les indices pertinents pour effectuer une mise en correspondance entre deux niveaux consécutifs. Quatre catégories d'indices ont été identifiées : les coins, les escaliers, les conduites et les murs porteurs. Un étage est donc représenté par un graphe de position relative entre ces indices. Pour reconstruire un bâtiment de plusieurs étages, une mise en correspondance entre les graphes correspondant à chacun d'eux est effectuée. Elle est basée sur l'utilisation d'un graphe de correspondance dans lequel la plus grande clique maximale est recherchée. Notons que des priorités sont affectées à chacun des indices pour cette étape de mise en correspondance.

En terme de stratégie, Dosch précise que l'enchaînement des traitements qu'il présente est dédié au type de plans traités. Il affirme toutefois que, de par la nature indépendante de ces traitements, l'enchaînement peut être modifié pour s'adapter facilement à d'autres types de représentation.

Néanmoins, il n'explicite pas réellement de modèle de ces traitements en terme de réglage de paramètres, ni de stratégie globale.

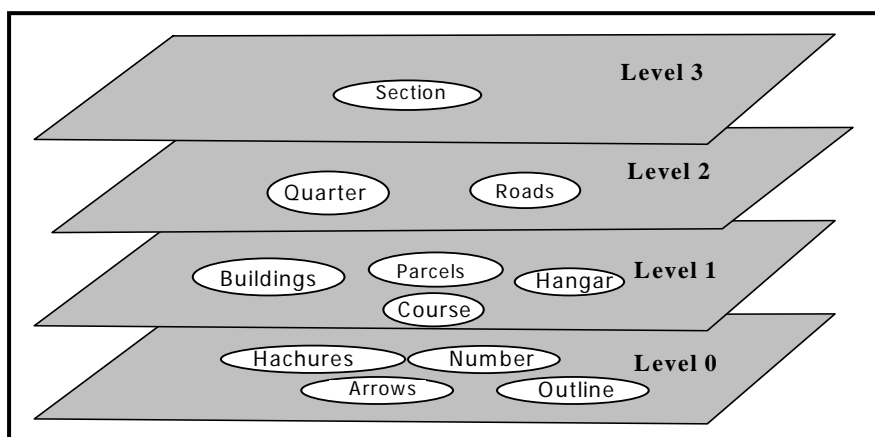
### **3.3.1.6. Le système SYRADOc de Ogier [Ogier 2000a]**

Le système SYRADOc est un système développé dans notre laboratoire pour l'interprétation des planches cadastrales françaises. En terme de connaissances, en adoptant la catégorisation précédente (descriptives / opératoires / stratégiques), les principales contributions de ce système concernent les connaissances descriptives et stratégiques, sous la forme, d'une part, d'un modèle de document (regroupant les objets pouvant y apparaître ainsi que leurs relations spatiales et sémantiques), et d'autre part, des plans de correction d'incohérences issues d'une première phase de traitement.

- Les connaissances descriptives

Les auteurs de ce système ont fait d'importants efforts en ce qui concerne l'explicitation des connaissances du domaine considéré par le document, ceci en vue de les extraire du code. Dans ce cadre, l'approche des auteurs a consisté, dans un premier temps, à lister l'ensemble des objets susceptibles d'apparaître sur les planches. Ces objets ont ensuite été répartis suivant 4 niveaux issus de l'observation des documents en cherchant à reproduire la méthodologie de conception de l'agent cadastral (Figure 3.6)

A titre d'exemple, cette Figure 3.6 montre qu'une parcelle est d'un niveau supérieur à celui des segments, des hachures ou encore des flèches, ce qui montre que ces derniers sont utilisés dans la construction des parcelles. Néanmoins, une telle hiérarchie des données ne suffit pas à elle seule à fournir un modèle du document qui doit être interprété. C'est pourquoi les auteurs intègrent aussi dans leur modèle un ensemble de règles retranscrivant les principes suivis par l'agent cadastral pour dessiner la planche. Ces règles sont intégrées dans une structure qui est représentée par le Tableau 3.1 et qui permet de ne pas faire apparaître le modèle du document dans le code de construction des objets.



*Figure 3.6 : Un modèle en couches des données « métier »*

- Les connaissances stratégiques

Dans ce système, une notion très intéressante nommée « cycles perceptifs » est introduite. Après une première phase de traitement, purement ascendante, qui permet de fournir les données du niveau 0 (Figure 3.6) associées à un niveau de confiance, un ensemble de constructeurs de données métier est déclenché, en vue de fournir une première version évaluée de l'interprétation (trois constructeurs sont utilisés : *parcelles*, *quartiers* et *routes*). C'est alors qu'intervient le « cycle perceptif » qui va permettre de gérer la cohérence des données reconstruites (Figure 3.7). Lors de l'exécution de ce cycle, les objets jugés incohérents à l'issue de la construction sont re-traités, chacun des types d'incohérence possible étant associé à une « solution remède » (Tableau 3.2). Une telle solution est une nouvelle chaîne de traitements appropriés, qui peut revenir le cas échéant sur les données images (une procédure de classification des incohérences est utilisée). Cette stratégie de bouclage permet aux auteurs un gain de 17 % en terme de taux de bonne classification.

<b>Object :</b>	<b>N(*)</b>	<b>Rules</b>	<b>Fct</b>	<b>Entity</b>		<b>N</b>	<b>Rules</b>	<b>Specification</b>	
Outline	=1	Included		vectors		> 3	Shared	essential	
Identifiant	=1	Included	Or	Toponym		≥1	Included	essential	
				Or	And	incoming arrow	= 1	Shared	essential
					And	Identifiant	=1	neighbor	
				Or	And	identifying arrow	= 1	Shared	essential
					And	Identifiant	=1	neighbor	
Hatching	=1	Included		2D Texture				optional	
Dashed line	≥1			1D Linear texture				optional	
Boundary mark	≥1			Symbol		≥1	Shared	optional	
Mail box number	≥1	neighbor		Toponym		=1	neighbor	Optional	
Out going arrow	≥1	Shared		Symbol		≥1		Optional	

*Tableau 3.1 : Le modèle d'une parcelle*

Une interaction est également présente dans ce système. L'utilisateur dernier peut intervenir de trois façons différentes : pour corriger les incohérences issues de la phase ascendante, pour contrôler l'autonomie du cycle perceptif, en déterminant le nombre de cycles autorisés et enfin pour apprendre au système de nouveaux plans de traitement des incohérences. Cette catégorisation des erreurs et le cycle de correction qui lui est attaché constituent les grandes originalités de ce système. En revanche, son défaut majeur est de ne pas modéliser explicitement les traitements utilisés dans les différents constructeurs de données, et ne pas fournir de mécanismes d'acquisition et d'évolution des différents types de connaissances qui ont été explicitées.

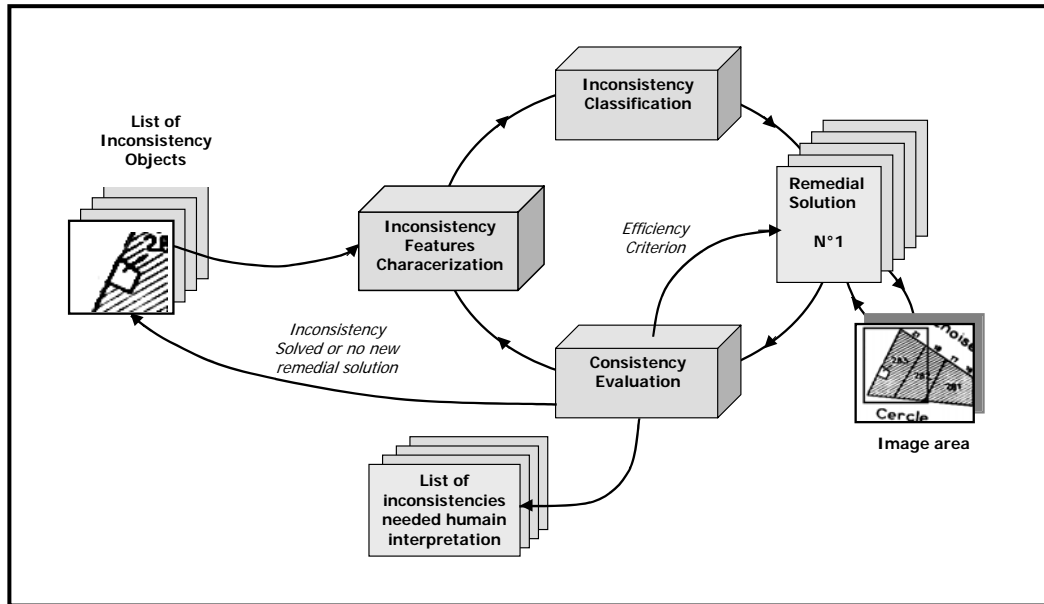


Figure 3.7 : Résolution des incohérences dans Syradoc : le cycle perceptif

Class Number	Inconsistency features	ClassicOrigin of Inconsistency
1	<b>Inconsistency features</b> <b>ClassicOrigin of Inconsistency</b>	No Identifier, No incoming vector Stroke of hatched area confused with parcel outline <b>OR</b> Character occlusions when character is connected to outline
	<b>Remedial solutions</b>	1- Parameters relaxation on Hatched extractor 2- Detection of characters connected to the outline
	<b>Frequency</b>	High
2	<b>Inconsistency features</b> <b>ClassicOrigin of Inconsistency</b>	No Identifier, many incoming vectors Part of hatched area not extracted (variability) <b>OR</b> character is connected to outline
	<b>Remedial solutions</b>	3- Parameters relaxation on Hatched extractor 4- Complementary Hatched area extractor algorithm 5- Detection of characters connected to the outline
	<b>Frequency</b>	Low
3	<b>Inconsistency features</b> <b>ClassicOrigin of Inconsistency</b>	Identifier, one incoming vector Character connected to parcel outline <b>OR</b> Arrow damaged during hatched area extraction
	<b>Remedial solutions</b>	6- Detection of characters connected to the outline 7- Arrow pattern matching without hatched area extraction 8- Parameters relaxation on Hatched extractor
	<b>Frequency</b>	Medium
4	<b>Inconsistency features</b> <b>ClassicOrigin of Inconsistency</b>	Identifier, included vectors Character connected together <b>OR</b> Symbols included in parcel <b>OR</b> Part of hatched area not extracted (variability)
	<b>Remedial solutions</b>	9- Detection of characters connected together 10- Detection of symbols included in parcel 11- Parameters relaxation on Hatched extractor
	<b>Frequency</b>	Medium

Tableau 3.2 : Table de résolution des incohérences



### **3.3.1.7. Le système CELESSTIN [Vaxivière 1994]**

Le système CELESSTIN est un système développé à l'ESSTIN par l'équipe de K. Tombre qui a pour but l'interprétation d'images de schémas mécaniques et leur conversion au format **CAD** (Computer Aided Design). Ce système a donné lieu à de nombreuses publications, au fur et à mesure des versions du système, mais [Vaxivière 1994] aborde plus particulièrement le problème de la représentation des connaissances que celles-ci soient structurelles, syntaxiques et sémantiques.

- Les connaissances descriptives

CELESSTIN est un système multi-spécialistes à base de *blackboards*. Les connaissances descriptives de ce système sont donc représentées dans l'architecture même de cette structure de stockage des données, qui est elle-même en adéquation avec l'organisation des spécialistes permettant l'évolution des données. Ces *blackboards* sont organisés en niveaux correspondant aux différents objets à extraire. Dans une couche donnée, un objet est représenté par un nœud avec lequel sont associées les informations appropriées : les attributs décrivant les primitives d'objets ainsi que des liens se référant aux relations inter objets. Il y a dans CELESSTIN IV, quatre principaux niveaux : les blocs (l'élément de base n'est pas le vecteur mais le « bloc », c'est à dire un polygone fermé), les « formes » qui sont en fait l'aspect global de la pièce dessinée, les entités symétriques et les entités fonctionnelles. Aux deux derniers niveaux sont intégrées ce que nous avons appelé « connaissances métier » dans la première partie puisque ici rentre en ligne de compte l'expertise du spécialiste « mécanique ». Ces connaissances « métier » sont intégrées procéduralement dans le système, sous la forme d'un ensemble de règles qui sont appliquées par trois spécialistes : le premier concerne la recherche d'entités symétriques, le second permet le désassemblage de la pièce globale en plus petites entités fonctionnelles, puis le dernier tente d'affecter une cinématique à chacune de ces entités.

- Les connaissances opératoires

Dans ce système, les traitements sont implémentés par l'intermédiaire de la partie action des spécialistes. Ceux-ci permettent l'évolution des données des différents *blackboards* et sont donc regroupés eux aussi en niveaux. Néanmoins, il ne semble pas y avoir dans ce système, de modélisation des traitements d'images et de leurs paramètres.

- Les connaissances stratégiques

Dans ce système, une stratégie de traitement se décompose en trois niveaux, comme le montre la Figure 3.8. Le bloc stratégie reçoit un résumé des états du *blackboard* et supervise l'ensemble du processus d'interprétation en déclenchant les tâches appropriées. Au moment de l'écriture de l'article, la stratégie est purement « linéaire » mais la structure de *blackboard* devrait, aux dires des auteurs, pouvoir supporter tous types de stratégie. Les tâches sont les sources de connaissance responsables du

contrôle. Leur but est de diriger et de coordonner l'action de divers spécialistes. Elles sont déclenchées par la stratégie et sont formées d'un ensemble de règles permettant l'activation des spécialistes. Elles reçoivent également les résultats des spécialistes déclenchés par l'intermédiaire d'événements. Enfin, les spécialistes accèdent directement au *blackboard*, et ils sont activés lorsqu'une partie pré condition est valide. Ils contiennent également une partie action qui peut être un programme, un système expert ou un ensemble de règles permettant de faire évoluer le contenu du *blackboard*.

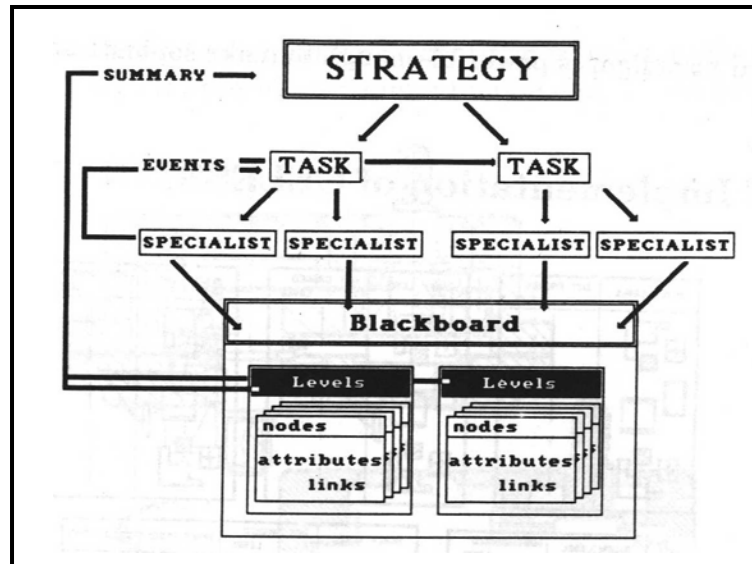


Figure 3.8 : Stratégie, tâches et spécialistes dans CELESSTIN

En guise de conclusion, les auteurs précisent que, même si la première stratégie implémentée est purement « linéaire », l'architecture choisie devrait permettre d'élaborer des chaînes de traitements plus sophistiquées. De plus, ils constatent également que les lacunes de leur système reposent essentiellement sur l'aspect procédural de la connaissance, ce qui entraîne des difficultés de gestion de la cohérence dans les *blackboards*. Ils proposent, dans ce cadre, l'utilisation de méthodes de représentation plus structurées et plus hiérarchiques, avec éventuellement l'utilisation de *frames*.

### 3.3.1.8. Le système de Ramel [Ramel 1998][Ramel 2000]

Le système de Ramel est un système développé par l'équipe **R**econnaissance de **F**ormes et **V**ision de l'INSA de Lyon. Il vise à reconnaître des documents techniques en s'inspirant des mécanismes de la perception humaine. L'application principale dans le cadre de ce travail est l'analyse de schémas cinématiques, mais selon les auteurs, elle est adaptable à d'autres types de documents (comme les planches cadastrales ou les partitions musicales). Quelques figures illustrent d'ailleurs certaines parties de l'approche avec d'autres catégories de documents. Comme pour [Joseph 1992][Den Hartog 1996] ou encore [Ogier 2000a], un soin particulier a été apporté dans ce système à la représentation des connaissances descriptives mais aussi stratégiques.

- Les connaissances descriptives

Deux niveaux peuvent être distingués en terme de description des objets : le premier est appelé « perception globale ». Il regroupe des objets bas niveau, dits structurants, qui permettent d'avoir une vision globale du document (l'expression « vision globale » étant utilisée en référence aux travaux sur la perception humaine). Ils sont utilisés, d'une part, afin de catégoriser les différents types de documents à traiter et, d'autre part, pour alimenter les processus de reconstruction des objets de plus haut niveau. Ces objets structurants sont au nombre de trois : le pixel, le vecteur et le quadrilatère. Le choix de ces primitives est fondé sur leur adéquation aux formes à décrire, mais aussi sur leurs aptitudes à représenter les connaissances acquises au cours de l'analyse. Une organisation hiérarchique de description est adoptée pour la représentation de ces objets. La Figure 3.9 représente ces objets suivant leur niveau d'abstraction et la quantité d'information qu'ils véhiculent. A chacune des primitives représentées sur cette figure correspond une classe d'objets, dans le but de faciliter non seulement la gestion des connaissances, mais aussi la vérification de la cohérence des résultats. Cela permet, à tous les niveaux de la hiérarchie, d'utiliser toutes les informations exprimées dans les niveaux inférieurs. Notons que chacune des classes est munie d'attributs pour gérer les connaissances, mais que ceux-ci ne sont pas décrits dans [Ramel 1998].

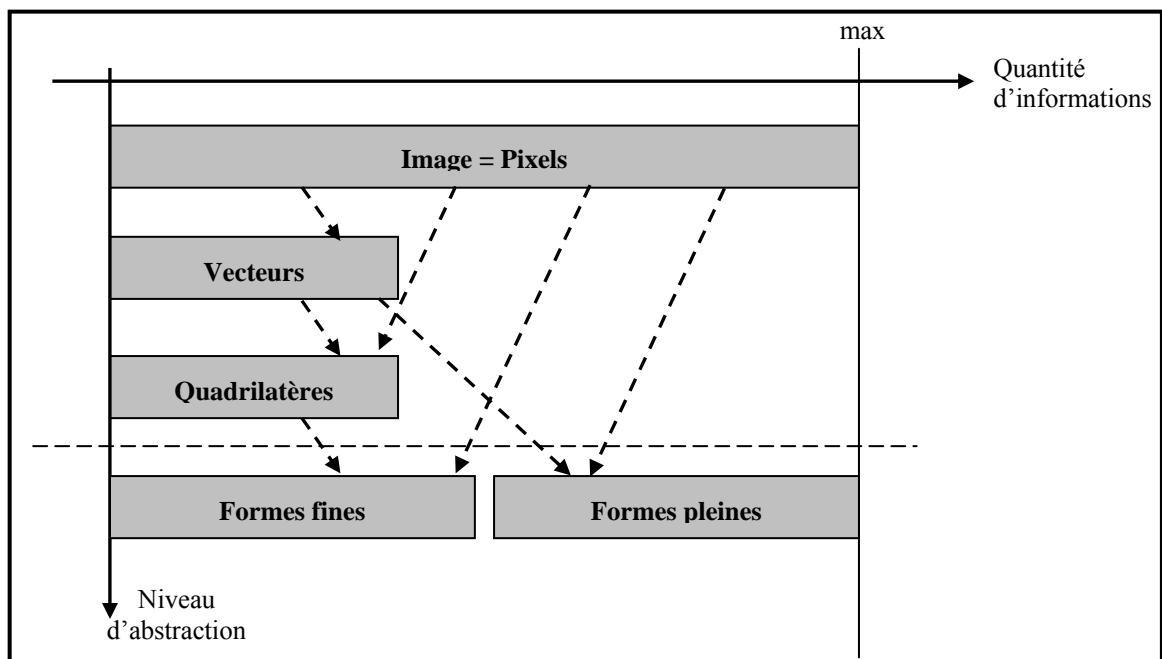
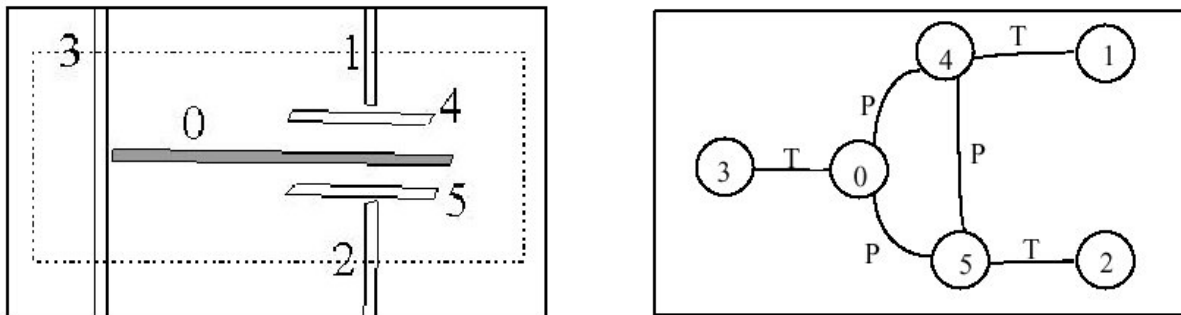


Figure 3.9 : Hiérarchie des classes d'objets dans [Ramel 1998]

Le second niveau est dit « de lecture et de compréhension ». Il a pour but d'intégrer les règles auxquelles sont soumis les dessinateurs de documents techniques. L'auteur souligne que c'est la connaissance de ces règles qui permet de vérifier la cohérence de la représentation obtenue, mais aussi de déterminer l'ordre d'extraction des différentes entités constituant un dessin. Ce niveau est basé sur la représentation de cinq types de données : le texte, les formes pleines, les lignes, les courbes et les

éléments mécaniques. Ces données sont construites sur la base des primitives du niveau « global ». Les quatre premiers types d'objets sont représentés procéduralement, par l'intermédiaire de spécialistes qui sont proches d'agents. Les éléments mécaniques sont eux aussi construits par l'intermédiaire d'agents, mais ceux-ci exploitent, pour ce faire, une représentation explicite des objets à extraire, sous forme de graphes. Ces graphes permettent de lier les différents objets qui constituent le dessin en fonction de leur voisinage. C'est donc à ce niveau qu'est intégrée la connaissance du domaine, en définissant l'agencement des objets sur le document. Dans un premier temps, le graphe construit ne contient que les quadrilatères, qui sont liés par les relations de liaison (en L ou en T), d'intersection (en X), de parallélisme ou de succession. La Figure 3.10 illustre un tel graphe pour un exemple simple. Puis, en utilisant un dictionnaire des entités mécaniques représentées sous forme de graphes, une étape de recherche de sous graphe commun, permet de faire apparaître progressivement les objets du domaine. Les objets du domaine sont donc explicitement représentés par ce dictionnaire.



*Figure 3.10 : Illustration des graphes représentant les connaissances du domaine dans [Ramel 1998]*

- Les connaissances stratégiques

En terme de stratégies, ce système adopte, comme [Ogier 1995], une approche à base de cycles perceptifs, inspirée de la perception humaine. Ce mécanisme permet de faire coopérer les différents spécialistes qui sont activés de manière séquentielle et bouclée, comme l'illustre la Figure 3.11. Une stratégie d'interprétation est donc représentée sous forme d'une boucle. Cette approche est très intéressante puisque les cycles permettent de relier les différentes perceptions et de combiner les opérations pour atteindre un but commun. Toutefois, si la structure de boucle semble adaptée, une partie de l'expertise est implicitement représentée dans le choix de l'ordre de déclenchement des différents spécialistes. Les auteurs soulignent que cet ordre doit varier en fonction du document, mais ne proposent pas de processus d'acquisition qui pourrait permettre d'apprendre l'enchaînement idéal. De plus, il peut paraître dommage de ne pas avoir inclus la première phase d'extraction d'une représentation de l'image sous forme de « primitives globales » dans cette structure de boucles.

Notons finalement que la représentation de connaissances opératoires est évoquée en perspective de ce travail.

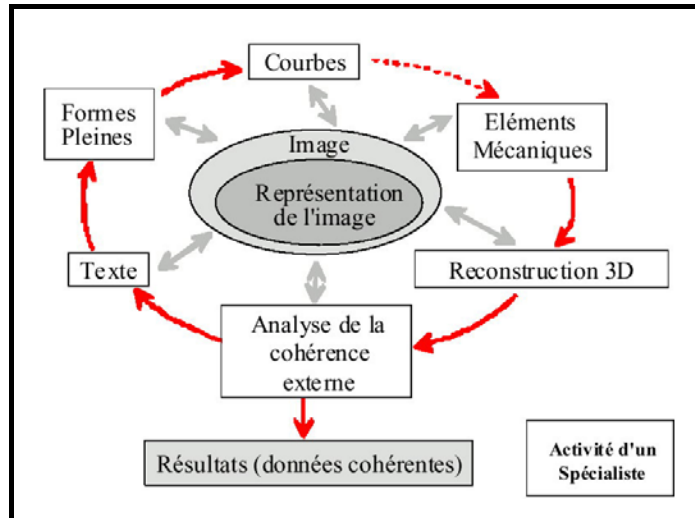


Figure 3.11 : Les connaissances stratégiques dans [Ramel 1998] : un cycle perceptif

### 3.3.1.9. Un bilan pour les systèmes d'interprétation de documents techniques

Dans le cadre de cette étude bibliographique, nous nous sommes intéressés à l'analyse de huit systèmes d'interprétation de documents techniques complets. Dans ces systèmes, les différentes équipes ont abordé le problème de la représentation des connaissances de façons différentes. Dans le Tableau 3.3, nous synthétisons les méthodes adoptées pour représenter les trois types de connaissances identifiés. Notons que concernant les différents opérateurs qui sont exploités dans une chaîne d'interprétation de documents techniques, le lecteur intéressé trouvera des états de l'art extrêmement complets dans les HDR [Tombre 1996], [Mullot 2000] et [Ogier 2000b].

	Connaissances descriptives	Connaissances opératoires	Connaissances stratégiques
Den Hartog	Réseaux sémantiques	Non explicitées	Réseau sémantique
Joseph	Réseau de <i>frames</i>	Fonction membres	Base de Règles
Pasternak	Langage de description	Exécutables issus du langage compilé	Mécanismes de trigger
Yu	Arbre de composants boucles et vecteurs	Non explicitées	Non explicitées
Ogier	Tableaux de description	Non explicitées	Cycles perceptifs et scénarii de résolution des incohérences
Ah Soon	Langage de description puis Réseau de contrainte	Non explicites	Non explicitées
Vaxivière	Pas de modèle du document explicite, il est ancré dans l'architecture du système	Non explicitées	Découpage en tâches qui sont réalisées par des spécialistes
Ramel	Graphes basés sur des primitives globales pour les connaissances du domaine	Non explicitées	Cycles perceptifs

Tableau 3.3 : Les connaissances dans les systèmes d'interprétation de documents techniques

*i. les connaissances descriptives*

Pour chacun des systèmes analysés, une réflexion a été menée concernant les connaissances descriptives. Excepté dans le cas de Den Hartog, qui exclut la notion de hiérarchie (mais qui le regrette), un consensus semble se dégager pour une hiérarchisation des entités à extraire, basée sur des relations de composition et de spécialisation. Cela permet indéniablement de sortir de l'effet « usine à gaz » d'une représentation « plane ». En fonction des systèmes, le nombre de « couches » dans la description hiérarchique de décomposition varie. Néanmoins, trois niveaux peuvent toujours être distingués, même s'ils sont fractionnés.

Au bas niveau, la plupart des auteurs considèrent comme entité de base les vecteurs (éventuellement les arcs) extraits d'une segmentation suivie d'une vectorisation. Selon nous, une telle démarche engendre une perte non négligeable des informations présentes sur l'instance papier. En effet, comme nous avons pu le voir dans la première partie du manuscrit, des vues graphiques autres que celles consistant à décomposer les formes en vecteurs et en arcs, peuvent s'avérer efficaces.

Dans les niveaux intermédiaires, les langages de description utilisés par Pasternak et l'équipe du LORIA, ainsi que les graphes de quadrilatères de Ramel sont particulièrement intéressants puisqu'ils permettent une définition explicite des entités à extraire. Cela fournit indéniablement une généralité aux systèmes. Néanmoins, force est de constater que dans chacun des cas, une maîtrise logicielle du système est nécessaire pour la définition d'un nouvel objet, l'accent n'ayant pas été mis sur l'aspect acquisition de connaissances.

Dans les couches les plus hautes, certains systèmes proposent une représentation des connaissances du domaine. C'est par exemple le cas de Den Hartog, dont le réseau sémantique « mono-couche » permet une adaptation rapide d'un domaine à un autre. Les tableaux de [Ogier 2000a] pour la représentation des objets du domaine est également intéressante. Néanmoins, ces tableaux sont totalement dédiés au problème traité. Une éventuelle modification des tableaux (que ce soit les objets ou leurs propriétés) n'est pas envisagée.

*ii. les connaissances opératoires*

Concernant les connaissances opératoires, force est de constater que bien peu d'équipes ont mis l'accent sur la modélisation des traitements inclus dans leur « bibliothèque image ». Aucun des systèmes précités n'aborde en effet l'adéquation traitement / contexte ou encore les méthodes de réglage des paramètres en fonction d'un objectif et d'un contexte donnés, si ce n'est en terme de perspectives. Les connaissances opératoires sont donc représentées de façon procédurale dans ces systèmes. Nous montrerons ci-après que quelques systèmes dédiés à l'interprétation de scènes ont abordé ce problème.

*iii. les connaissances stratégiques*

Concernant les connaissances stratégiques, les différents systèmes ont des approches relativement différentes, mais chacune semble apporter un « plus » pour la résolution du problème d'interprétation. Parmi les points à souligner dans ces systèmes, citons l'exploitation des relations spatiales entre objets pour la génération automatique de *scenarii*, dans [Den Hartog 1996]. Ce point, en étant généralisé à d'autres types de relations, ouvre des perspectives tout à fait intéressantes pour l'adaptabilité du système. Les règles définies dans Joseph rejoignent cette réflexion. Il est toutefois regrettable que les exécutions de ces règles ne soient pas contrôlées par un mécanisme de niveau supérieur (des méta règles par exemple). Une autre piste intéressante est le mécanisme de décomposition de la tâche initiale en sous tâches, présent dans [Vaxivière 1994], qui permet d'éviter l'effet « usine à gaz ». Enfin, les mécanismes de cycles perceptifs [Ogier 2000a][Ramel 1998] et de gestion des incohérences [Ogier 2000a], semblent également apporter une fiabilité accrue au système.

En guise de conclusion, l'existence de ces systèmes constitue une preuve indéniable que la communauté du document technique commence à se tourner vers des problématiques orientées ingénierie des connaissances. Les différents formalismes existant en Intelligence Artificielle (règles, *blackboards*, *frames*, réseaux sémantiques, systèmes multi-agents ...) sont désormais exploités pour la représentation de connaissances dans le contexte de l'interprétation de documents techniques. Toutefois, force est de constater que la réussite de ces systèmes est pour l'instant souvent limitée à l'application pour laquelle ils ont été conçus. Selon nous, ceci est sans conteste dû à l'aspect encore trop peu explicite de la représentation des connaissances, qui empêche l'intégration simple de nouvelles connaissances et leur manipulation. En effet, si chacun des systèmes analysés apporte une contribution dans ce sens, il n'existe pas encore de système gérant globalement l'ensemble des aspects modélisation, représentation, acquisition, opérationnalisation et évolution des connaissances évoquées dans le premier chapitre de cette partie. Dans la section qui suit, nous abordons cette problématique dans le cadre plus général des systèmes de vision.

### **3.3.2. Dans les systèmes de vision en général**

Au vu du peu de littérature abordant l'interprétation de documents techniques comme un problème d'acquisition, de représentation puis de manipulation de connaissances, il nous a paru opportun d'analyser la bibliographie qui concerne les systèmes de compréhension de scènes. En effet, ces systèmes bénéficient d'un historique plus important et se sont donc tournés avant la communauté du document vers des aspects ingénierie des connaissances. Dans cette section, nous proposons d'abord une synthèse des formalismes de représentation ayant été adoptés dans la littérature de ce domaine, sans toutefois être exhaustif. Puis, nous décrivons trois systèmes particuliers qui ont inspiré notre

approche pour le projet *NATALI* en proposant des réponses intéressantes aux questions de l'acquisition, de la modélisation et de la représentation des connaissances.

### **3.3.2.1. Les formalismes utilisés dans les systèmes de vision**

Les états de l'art dressés dans [Mastuyama 1993][Sandakly 1995][Crevier 1997] ou [Ogier 2001] montrent que dans le domaine de l'interprétation de scènes, comme pour l'interprétation de documents techniques, la plupart des formalismes utilisés en intelligence artificielle peuvent être adoptés. Dans ce paragraphe, nous reprenons chacun de ces formalismes, en citant pour chacun quelques systèmes concernés et en analysant les avantages et les inconvénients des différentes approches. Cette liste est présentée en suivant l'évolution historique des systèmes qui a globalement suivi les grandes tendances de l'Intelligence Artificielle.

#### *i. Les systèmes à base de règles*

Avec les premiers résultats de l'Intelligence Artificielle, les premiers systèmes de vision à base de connaissances pouvaient être vus comme des systèmes de production, à base de règles. A notre connaissance, le premier exemple à avoir adopté ce formalisme des règles est le système développé par l'équipe de Levine [Levine 1978, 1985]. Les règles y sont utilisées pour représenter les connaissances au niveau de la scène (le domaine dans la terminologie utilisée précédemment). Ces règles floues, qui décrivent le monde, permettent l'utilisation d'une approche descendante, guidée par les buts. Ce système exploite en outre deux bases de données relationnelles pour représenter les connaissances descriptives. La première (**Mémoire à Long Terme MLT**) permet la description *a priori* des objets à extraire, de leurs propriétés (dont les valeurs moyennes sont stockées, ce qui peut sembler un peu limité) et de leurs relations spatiales. La seconde (**Mémoire à Court Terme MCT**) sert quant à elle de mémoire de travail en stockant les résultats de l'interprétation en cours. Dans ces bases, les données sont hiérarchisées suivant trois niveaux d'abstraction : le bas niveau (image), le niveau intermédiaire (contours/région) et le haut niveau (objets). A chacun des niveaux de la **MCT** sont présentes les instances du modèle défini dans la **MLT**, avec des valeurs affectées aux attributs et aux relations spatiales. Une interprétation associée à chaque région et un historique des traitements effectués sont également intégrés à cette **MLT**. Notons qu'une originalité de ce système est de décrire les objets non seulement par une liste de propriétés, mais aussi en y faisant figurer les propriétés particulières qui permettront de discriminer un objet donné

Le formalisme des règles est également à la base du système **ACRONYM** de Brooks [Brooks 1981, 1983]. Ce dernier est l'un des premiers systèmes dont la vocation est d'être générique, en ayant donc pour ambition d'être indépendant du domaine d'application. Toutefois, il n'a concrètement été testé que sur des images d'aéroport. Dans ce système, les objets sont modélisés en 3-D et détectés sur des images 2-D. Les apparences géométriques des objets dans la scène sont modélisées par l'intermédiaire



des primitives les constituant, et de leurs relations spatiales. Elles sont représentées sous forme de 280 règles de production dans un module appelé module de prédiction. En 3-D, les descriptions de ces objets sont effectuées à l'aide d'*expressions* qui sont des combinaisons de variables prédéfinies utilisant des opérateurs tels que +, -, \*, < et >. Notons aussi qu'une modélisation hiérarchique des objets exploitant des liens de type « partie-de » ou « sorte-de » est intégrée. Elle est effectuée en utilisant le formalisme des *frames* de Minsky [Minsky 1975] et de cônes généralisés.

Le système SPAM de McKeown [McKeown 1985, 1989] exploite également le formalisme de règles. Celui-ci avait initialement pour but l'interprétation de scènes d'aéroports, mais a été étendu pour l'interprétation de zones suburbaines grâce à la mise en œuvre d'un module d'acquisition de connaissances et d'un module d'évaluation de l'interprétation. La modélisation des connaissances a fait l'objet d'un gros effort. Les objets sémantiques sont d'abord représentés par le formalisme des *frames*, qui sont ensuite transformées, par l'utilisation d'un module nommé RULEGEN, en un ensemble de règles. L'acquisition de cette connaissance sous forme de *frames* peut se faire de deux façons, soit par un éditeur de *frames* en ligne, soit par l'exemple. Une centaine de *frames* est utilisée, générant chacune 5 règles. Notons que le module d'acquisition (ISCAN) est remarquablement pensé car il permet une interaction entre le système et l'utilisateur. Ce dernier peut ainsi définir les objets du domaine considéré. Nous verrons que cette approche a inspiré nos travaux pour *NATALI v2*.

Tous ces systèmes à base de règles ont globalement montré l'intérêt d'une approche à base de connaissances. Toutefois, ils sont, au même titre que les systèmes de production en Intelligence artificielle, limités par leur aspect monolithique et leur non-évolutivité.

## **ii. Les systèmes à base de Tableaux noirs**

Pour pallier ce monolithisme des approches à base de règles, les équipes travaillant sur le problème de l'interprétation de scènes ont parfois adopté des approches à base de Tableaux Noirs (*Blackboards*) [Hayes 1985]. Une telle approche utilise un ensemble de sources de connaissances, le tableau noir proprement dit et un module de contrôle. Les sources de connaissances possèdent une certaine compétence et ont une vue réduite sur l'état de la solution en cours d'élaboration dans le tableau noir. Le tableau noir est une sorte de structure de données qui permet de mémoriser l'état de la solution résultant de l'application des différentes sources de connaissances. Le module de contrôle supervise le déclenchement des différentes sources de connaissances.

Ces tableaux noirs sont utilisés dans [Levine 1981], qui propose un système de segmentation d'image exploitant les connaissances du domaine (des scènes d'extérieur). Dans ce système, différents « *traiteurs* » (les sources de connaissances dans la terminologie des *Blackboards*) coopèrent pour

instancier des objets dans le tableau noir. Les *traiteurs* ne communiquent pas entre eux mais au moyen de la mémoire partagée qu'est le tableau noir.

Le système SACAP, développé par Nagao et Matsuyama [Nagao 1979, 1980], s'appuie lui aussi sur ce principe des tableaux noirs. Toutefois, ce système, dédié à l'interprétation d'images aériennes, n'a pas fait l'objet d'efforts en ce qui concerne la modélisation des connaissances. Tous les objets, les propriétés et les relations entre objets sont codés sous forme procédurale, dans les sources de connaissances, ce qui rend le système non évolutif et inexplicable. Aucune distinction n'est faite entre les objets, leur rôle dans la scène et leur représentation dans l'image. De plus, aucune hiérarchie n'est exploitée.

En guise de bilan, ces systèmes à base de tableaux noirs apportent indéniablement plus de modularité, ce qui permet une modification plus facile du système. Toutefois, une telle approche pose le problème du contrôle qui reste centralisé. Avec le grand nombre de sources de connaissances qui doivent être mises en œuvre dans un processus d'interprétation, un tel contrôle devient rapidement problématique et le système se trouve confronté au problème classique du « goulot d'étranglement ». Un autre problème qui est souligné dans [Crevier 1997] est celui de la temporalité des données. Les *blackboards* ont à l'origine été imaginés pour l'interprétation de la parole. Ils étaient donc basés sur un déclenchement automatique des sources de connaissances lors de l'apparition d'un mot, puis ces mêmes sources de connaissances disparaissaient. Dans le cas de l'interprétation d'image, les sources de connaissances correspondent généralement aux objets à reconnaître, il ne faut donc pas qu'elles disparaissent tant que l'intégralité des données n'a pas été extraite.

### ***iii. Les systèmes à base de frames et de réseaux de frames***

La notion de *frame* a été introduite par les travaux de Marvin Minsky [Minsky 1975]. Une représentation à base de *frames* est inspirée du fait que la réaction de l'être humain se base sur des représentations génériques de situations auxquelles il a déjà été confronté. En fonction des spécificités de la situation présente, l'Homme affine sa perception de la scène pour réagir de façon plus appropriée. Une *frame* est un cadre qui correspond à une situation déjà rencontrée. Elle est composée de *slots*, qui servent à la caractériser. Ces *slots* fournissent des informations sur la hiérarchie des *frames* (sorte de, est un), sur leurs propriétés (sa couleur...), sur leur rôle et sur les relations les unissant (près de, au-dessus...). Ils sont associés à des facettes qui permettent de caractériser la nature et le comportement d'un attribut et qui décrivent par exemple le type et le domaine de validité du *slot*. De par leur définition, les *frames* sont proches des classes au sens du paradigme objet. Toutefois, elles en diffèrent par le fait que les *frames* implémentent un langage représentationnel, alors que nous

voyons plutôt l'objet comme un langage de programmation, donc beaucoup plus rigide. En vision, les *frames* ont été largement utilisées.

C'est par exemple le cas dans le système VISIONS de Hanson [Hanson 1978, 1987], même si les *frames* ne sont pas explicitement nommées. Ce système (qui est probablement celui ayant fait l'objet du plus de recherches) adopte le découpage en trois niveaux d'abstraction du système de Levine : le bas niveau, qui considère les informations de type pixel, le niveau intermédiaire, qui prend en compte les primitives symboliques extraites à partir de l'image, et le haut niveau qui traite de l'information sémantique. A ce dernier niveau, un réseau est utilisé pour représenter la connaissance concernant les objets de la scène. A chaque nœud de ce réseau se trouve une structure nommée *schéma*. Un *schéma*, (qui peut être vu comme *une frame*), est une structure contenant toute la connaissance nécessaire à la reconnaissance d'un objet. Il peut aussi représenter un concept de haut niveau, comme une scène, un objet sémantique ou une sous partie d'un objet. Dans les *schemas*, deux catégories de connaissances sont représentées : les connaissances descriptives et les connaissances stratégiques. Les premières décrivent l'apparence des objets en prenant en compte le contexte (capteurs, échelles...). Les secondes concernent les stratégies impliquées dans la reconnaissance des objets, la vérification de leurs relations spatiales et la résolution des conflits avec d'autres instances de schémas.

Dans [Toriu 1987], les *frames* sont également utilisées, mais pour la représentation de connaissances opératoires. Les *frames* permettent dans ce système de représenter les entrées/sorties des traitements, leur contexte d'utilisation. Ces traitements sont organisés en arbre proche d'un principe Tâche / Méthode / Outils [Ficet 2000] et les *frames* assistent leur sélection.

Les *frames* peuvent donc être utilisées pour la représentation des connaissances descriptives, opératoires et stratégiques. Elles permettent même souvent de représenter l'ensemble de ces connaissances dans un formalisme unifié. Toutefois, un tel « mélange » des connaissances tend à masquer le comportement du système et à le rendre relativement incompréhensible.

#### ***iv. Les systèmes à base de réseaux sémantiques***

Les réseaux sémantiques constituent un autre formalisme parfois adopté pour la représentation des connaissances dans les systèmes de vision. De tels réseaux sont des graphes acycliques dans lesquels les nœuds représentent des concepts et les liens des relations entre ces concepts. Les nœuds comme les liens peuvent être associés à des relations appelées attributs. Ils constituent donc un formalisme assez naturel pour la représentation des connaissances descriptives.

L'un des premiers systèmes à utiliser ce formalisme est le système ERNEST de Niemann [Niemann 1990]. Ce système est le seul, à notre connaissance, ayant pour vocation d'interpréter à la fois parole et

image. Des applications concernant ces deux types de problèmes ont été montrées (reconnaissance de l'Allemand pour la parole et de scènes industrielles pour les images). La base du modèle dans ce système est un réseau sémantique modifié. Les nœuds y sont des structures de données complexes. Trois types de nœuds sont définis : les concepts, les concepts modifiés (incomplets), et les instances. Cinq types de liens différents sont intégrés : les classiques « partie de », « spécialisation de » et « instance de » mais aussi « concrétisation de » et « modèle de ». Ces deux derniers, moins classiques que les autres, expriment les relations entre différents niveaux d'abstraction des concepts. En terme de stratégie, l'utilisateur choisit un nœud dans le réseau et celui-ci devient alors le but à atteindre. Les nœuds voisins sont alors recherchés, que ce soit au même niveau ou aux niveaux inférieurs et un algorithme de type A\* permet d'ordonner les buts à atteindre. Notons que dans ce système, aucun formalisme de description de l'apparence graphique des objets n'est intégré.

Plus récemment, le système AIDA [Tönjes 1999][Liedtke 1997], développé à l'université de Hanovre, a lui aussi opté pour une représentation des connaissances par un réseau sémantique. Dans ce réseau, les nœuds sont les objets de la scène. Les liens dans le réseau sont de deux sortes : les liens hiérarchiques et les liens topologiques. Les liens hiérarchiques incluent les liens de spécialisation (sorte-de), de constitution (partie-de), de concrétisation (concrétisé-par) et des liens dits *data-of*. Ce dernier type de lien constitue la grande différence avec le réseau d'ERNEST. Il permet en effet de faire le lien entre les concepts initiaux et leur représentation graphique en terme de primitives. Les liens de type *data-of* font l'interface entre l'aspect symbolique du réseau et l'image numérique. Notons que les liens de type topologique permettent quant à eux de définir les relations spatiales entre les objets, transversalement aux couches formées par les liens hiérarchiques.

Les approches à base de réseaux sémantiques semblent, grâce à leur aspect graphique, particulièrement adaptées pour une représentation déclarative des connaissances descriptives des systèmes. Toutefois, dans les systèmes évoqués, les moteurs d'interprétation associés à cette connaissance explicitée semblent monolithiques et le contrôle est alors problématique.

#### **v. Les systèmes à base d'agents**

C'est pour répondre aux problèmes de contrôle évoqués aussi bien dans le cas des *blackboards* que dans le cas des réseaux sémantiques que les équipes se sont souvent tournées vers le paradigme agent. Un agent est une entité logicielle ou matérielle plus ou moins autonome travaillant dans un environnement partagé, dynamique et ouvert : le système multi-agents. De l'intelligence artificielle, la communauté est passée avec ces systèmes à l'intelligence artificielle distribuée, en « répartissant » les connaissances du système. La clef de voûte de tels systèmes est le double mécanisme d'inférence dont disposent les agents, à savoir le niveau résolution de tâches et le méta-niveau leur permettant de s'observer et d'interagir durant la résolution du problème. Pour l'interprétation d'images, les

approches à base d'agents sont extrêmement nombreuses dans la littérature, nous n'en citerons donc que quelques exemples représentatifs. Les agents dans ces systèmes peuvent être de deux types : les objets recherchés ou les traitements d'images.

Le système *SCHEMA* décrit dans [Draper 1989], est un exemple de systèmes dans lequel un objet de l'image est aussi un agent. Chacun des *schemas*<sup>12</sup> (le terme agent n'est pas utilisé par l'auteur, mais un *schema* peut être assimilé à un agent) correspond, en effet, à une hypothèse de présence d'objet dans l'image. Lorsqu'il est activé, l'agent a alors en charge de vérifier cette hypothèse, en mettant en œuvre une stratégie de traitement d'images. La stratégie globale du système est donc fonction du contrôle de chacun des agents. Les agents activés peuvent entrer en compétition pour s'attribuer des zones de l'image, ce qui rend les résultats obtenus plus fiables.

Le même genre d'agent est utilisé dans le système SIGMA décrit dans [Matsuyama 1990]. Les agents représentent les parties d'objets. Le système global est donc un arbre, basé sur la relation de composition. Lorsqu'un objet est instancié en utilisant des traitements d'images, son « père » génère une liste d'hypothèses qui correspondent aux « frères » de l'objet instancié (les autres constituants de l'objet père). Ces hypothèses deviennent à leur tour des agents. Chacun des agents dispose donc de connaissances descriptives, opératoires et stratégiques.

Concernant les travaux où les agents sont des traitements, de très nombreuses références existent. Citons les travaux de Watanabe [Watanabe 1990], qui propose une approche multi-agents permettant la coopération et l'intégration de trois algorithmes de *stéréo matching*. Différents agents hiérarchisés sont appliqués sur les deux images et ils négocient et fusionnent leurs résultats respectifs, au travers d'un protocole. En cas d'ambiguïté pour la prise de décision, ce protocole de négociation entre les agents permet d'inférer sur leur paramétrage respectif.

Les travaux décrits dans [Eallet 2000] exploitent également le formalisme d'agents. Le système décrit vise à détecter, reconnaître et identifier des cibles sur des images. Pour ce faire, cinq catégories d'agents sont exploitées dans le système : les agents de localisation, de détection, de focalisation, de mouvement et de fusion. De plus, un agent « concept » permet le contrôle des autres agents. Tous ces agents partagent les nombreuses connaissances nécessaires à la tâche d'interprétation. Ces connaissances sont stockées dans une base commune, fortement structurée et divisée en trois parties : la spécification du domaine de l'image, l'objectif initial du système et les concepts qui peuvent être trouvés sur l'image. Ces derniers sont associés à une description des méthodes correspondantes pour chaque catégorie d'agent. Typiquement, à un concept « route » seront associées une ou plusieurs méthodes de localisation, de détection... Les résultats obtenus par ces agents sont stockés dans une

structure appelée modèle du monde, qui contient une description hiérarchique de la scène, décrite sous la forme d'un réseau Bayésien.

Des travaux de l'équipe TCI de l'IRIT de Toulouse portent également sur l'interprétation d'images au travers d'approches multi-agents [Bessetes 1998][Desachy 1999]. Leur objectif est de faire coopérer différents opérateurs afin de pouvoir générer une image de classes ou d'obtenir une classification de meilleure qualité. Des méthodes de coopération adaptées au type d'images étudiées et aux opérateurs sont utilisées. Le cadre d'application envisagé concerne, par exemple, la classification de parcelles agricoles, la détection de zones urbaines ou la détection de routes.

Les avantages apportés par une approche à base d'agents sont désormais bien connus. Les plus évidents sont la facilité de construction et de maintenance, la possibilité de bénéficier des architectures parallèles, la capacité de focalisation et évidemment la fiabilité accrue apportée par les possibilités de coopération et de compétition (la co-exploitation au sens large). Toutefois, l'utilisation d'agents apporte souvent un aspect procédural à la représentation des connaissances. Les connaissances sont distribuées, mais sont souvent intégrées dans le code des agents. Un changement d'application ou l'ajout de méthodes n'implique alors pas une modification totale du système, mais il implique tout de même la construction de nouveaux agents, ce qui requiert des compétences en génie logiciel.

Après cette description des différents formalismes exploités dans le domaine de la vision, nous proposons ci-après une analyse plus fine de trois systèmes récents qui ont guidé nos travaux pour le projet *NATALI*.

### **3.3.2.2. Les systèmes à la base de notre approche**

Trois systèmes de vision de la littérature ont inspiré notre approche dans le cadre du projet *NATALI* et méritent donc une description plus détaillée. Cette analyse est présentée dans ce paragraphe.

#### *i. Le système MESSIE II de Sandakly [Sandakly 1995]*

Le système *MESSIE II* (**M**ulti **E**xpert **S**ystem for **S**cene **I**nterpretation and **E**valuation) est un système d'analyse de scènes développé par une équipe de l'INRIA. Les domaines d'application visés sont la reconnaissance de scènes 3D d'intérieur et l'analyse d'images satellite. Les travaux concernant ce système ont été abordés dès 1988. Ils ont donné lieu à une première implémentation nommée *MESSIE I* [Montesinos 1990][Garnesson 1991]. Puis, dans le cadre de sa thèse, Sandakly a continué ce travail en identifiant les problèmes majeurs et en y apportant des solutions. Nous nous contenterons ici de décrire l'approche de la seconde version *MESSIE II* en terme de connaissances. Celle-ci est en effet très proche des préconisations de Newell. Elle a consisté à :

---

<sup>12</sup> A ne pas confondre avec les *schemas* du système *VISION* qui eux sont proches des *frames*

- Définir les connaissances nécessaires pour l'interprétation d'une scène (objets et mécanismes de raisonnement), et séparer la partie générique de la partie dédiée à l'application.
- Définir les formalismes de représentations de ces connaissances en fonction de leur type, en prenant en compte dans ces formalismes les notions d'incertitude et d'imprécision
- Choisir l'architecture logicielle appropriée pour intégrer ces connaissances, et les implémenter
- Tester dans le cadre d'applications réelles.

Dans ce système, les connaissances sont réparties suivant trois niveaux hiérarchiques :

- Le niveau de la scène (haut niveau), qui contient l'ensemble des objets sémantiques de la scène, leurs relations spatiales et une stratégie générique d'interprétation s'appuyant sur ces objets.
- Le niveau des objets sémantiques (niveau intermédiaire). A ce niveau, les stratégies de détection d'objets sont indépendantes et gérées par des spécialistes d'objets sémantiques. Au moins un spécialiste est associé à chaque objet, les stratégies de détection d'un objet peuvent être multiples.
- Le niveau extraction de primitives (bas niveau) : à ce niveau sont gérées les stratégies d'adaptation et de paramétrage des algorithmes de traitement d'image, en fonction de contraintes appliquées aux traitements. Néanmoins, l'auteur précise qu'une telle tâche de modélisation des paramètres pour la prédiction des résultats est très difficile. Il a donc essentiellement géré des contraintes spatiales et sur le type d'objet à extraire pour optimiser les résultats de ses algorithmes.

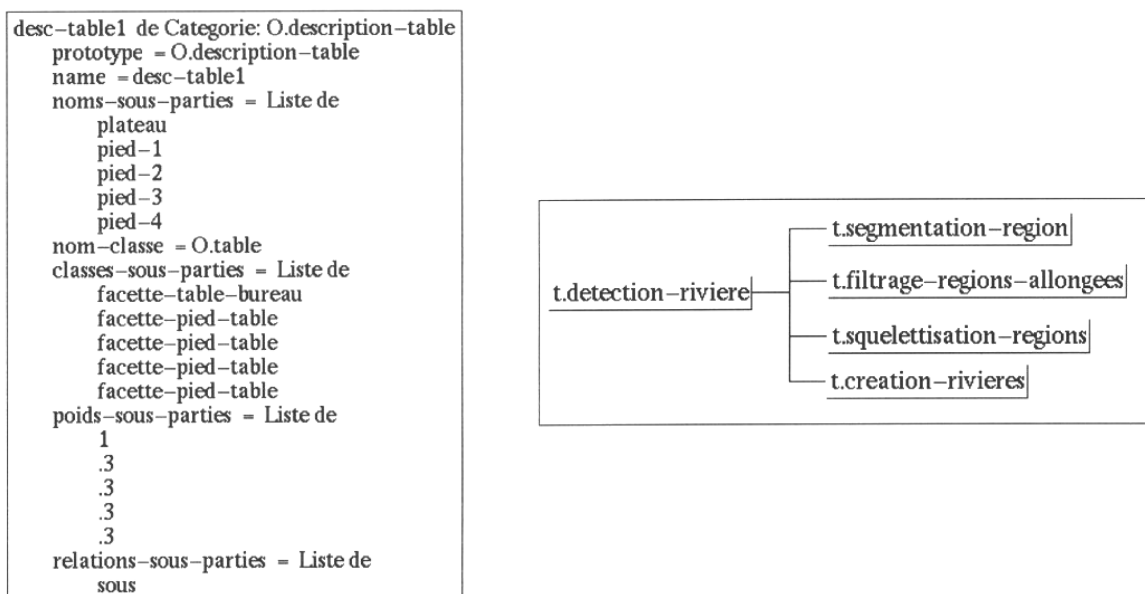
Dans la suite de cette description, nous nous attachons à examiner les connaissances intégrées dans chacun de ces niveaux, en reprenant la catégorisation proposée en 3.2.1.

- Les connaissances descriptives

Parmi les concepts les plus importants intégrés dans ce système, la modélisation des objets sémantiques joue un rôle considérable. Ceux-ci sont représentés par quatre points de vue différents :

- ❑ La géométrie : il s'agit de la décomposition des objets en sous-objets géométriques.
- ❑ La radiométrie : ce sont des adjectifs « flous » déduits des matériaux constituant les objets.
- ❑ Le contexte spatial : c'est l'ensemble des relations spatiales entre objets. Celles-ci sont décrites sous une forme déclarative par des propositions.
- ❑ La fonctionnalité : elle est décrite sous forme d'un symbole et est utilisée pour la catégorisation des objets sémantiques par la stratégie d'interprétation.

En terme de représentation, ce système utilise les capacités de SMECI (générateur de systèmes experts) pour représenter les objets. Deux formalismes de représentation sont proposés par ce générateur. Les catégories sont équivalentes aux *frames* et permettent une représentation hiérarchique des concepts avec des liens d'héritage de type « sorte de ». Il est possible d'y attacher des procédures par l'intermédiaire de *démons*, de valeurs actives et de méthodes déclenchées par des messages. Un exemple de catégorie permettant la représentation d'un concept table est proposé sur la Figure 3.12 à gauche. Chaque catégorie est associée à un prototype qui en définit les intervalles de valeurs possibles et certaines valeurs par défaut. Une base de règles est également utilisée. Elle contient des règles de production du premier ordre qui sont appliquées en chaînage avant. Un ensemble de règles forme alors une « tâche » qui elle-même se décompose fonctionnellement en arbre de sous-tâches. Ces tâches sont enchaînées dans ce système par l'intermédiaire des spécialistes d'un *blackboard* (NMS : **N**oyau **M**ulti **S**pécialiste [Corby 1992]) dont nous n'aborderons pas ici le fonctionnement. Un exemple de décomposition de la tâche « *détection\_rivière* » est proposé sur la Figure 3.12 à droite.



*Figure 3.12 : illustration des formalismes proposés par SMECI et utilisés dans MESSIE II*

- Les connaissances opératoires : le système MESSIE est l'un des rares, à avoir mis l'accent sur la modélisation des connaissances opératoires. Des interfaces « intelligentes » ont été développées ; elles ont pour but de paramétrer les opérateurs de traitement d'images, de filtrer leurs résultats et permettent de commander les capteurs (dans une philosophie « vision active »). Néanmoins, pour régler les paramètres, ces interfaces ne prennent en compte que des contraintes spatiales et le type des primitives à extraire. Le contexte de l'image reste inexploité.
- Les connaissances stratégiques : en terme de stratégie, les auteurs de ce système distinguent deux principales catégories : les stratégies de détection et les stratégies d'interprétation. En ce qui

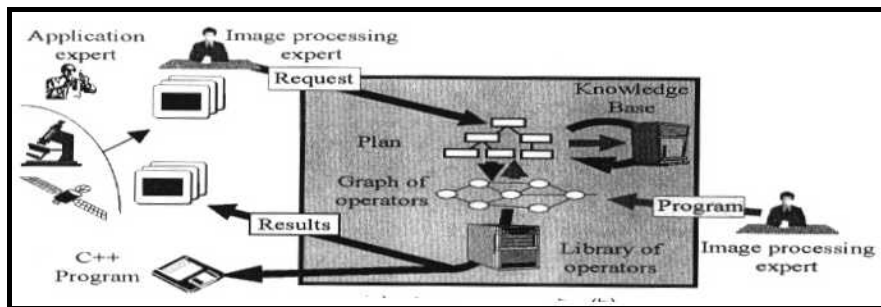


concerne les stratégies d'interprétation, une approche originale inspirée du processus d'analyse consciente humain est adoptée par cette équipe. Elle consiste à reconnaître dans un premier temps les objets les plus évidents qui représentent concrètement les grandes structures de la scène, plus faciles à identifier, le degré de facilité étant déduit de la géométrie et de la radiométrie des objets prototypés.

Ce système a beaucoup inspiré notre travail, essentiellement au travers de la démarche adoptée. En effet, la définition des connaissances nécessaires pour l'interprétation d'une scène (objets et mécanismes de raisonnement), et la séparation de la partie générique et de la partie dédiée à l'application semble incontournable lorsque le but est l'obtention d'un système générique.

*ii. Le système BORG de Clouard [Clouard 1999]*

Le système BORG, développé par Régis Clouard [Clouard 1999] au GREYC de Caen, a pour objectif la supervision d'une librairie d'opérateurs de traitement d'images (PANDORE [Clouard 1995]). Le but de ce système n'est pas d'interpréter les images, mais de construire, en fonction des désirs d'un expert, une chaîne d'opérateurs pouvant traiter toutes les images d'une classe donnée afin d'en extraire des informations au niveau image (compter le nombre d'objets d'une taille donnée par exemple)



*Figure 3.13 : Vision globale du système BORG [Clouard 1999]*

Ce système peut donc être vu, d'une part, comme un outil d'expérimentation (Figure 3.12 gauche), permettant à deux experts (l'un dans le domaine particulier de l'image traitée, l'autre en traitement d'images) de collaborer pour construire une chaîne d'opérateurs de traitement d'images. D'autre part, il peut être considéré comme un outil d'acquisition de connaissances (Figure 3.12 droite). Il est en effet aisé, grâce à ce système, d'ajouter de nouveaux opérateurs, de nouvelles tâches et donc un savoir-faire supplémentaire en traitement d'images. Pour satisfaire ces deux objectifs, le système BORG considère le développement d'applications de traitement d'images comme un problème de supervision d'opérateurs. Dans un tel contexte, les activités suivantes sont mises en jeu dans le système : sélection des opérateurs adéquats, détermination des paramètres, ordonnancement des opérateurs en chaînes, génération du script correspondant, exécution de celui-ci et enfin contrôle des résultats. En terme de connaissance, une telle approche privilégie donc clairement les connaissances opératoires et

stratégiques vis à vis des connaissances descriptives qui ne sont pas explicitées du fait de l'abstraction qui est faite des objets du domaine.

Dans l'article décrivant le système, deux bases de connaissances distinctes sont identifiées, chacune composée de sources de connaissances puisqu'une architecture de *blackboards* est adoptée. La première de ces bases est dédiée au problème particulier de la résolution de problèmes de traitement d'images alors que la seconde est dédiée au contrôle dans la résolution de problèmes en général. Par souci de clarté, nous gardons toutefois notre catégorisation opératoire / stratégique, même si nous inversons l'ordre de présentation de ces deux catégories pour des raisons que nous expliciterons.

- Les connaissances stratégiques : des efforts ont été réalisés dans la conception de BORG du point de vue des connaissances stratégiques. Dans la suite, nous analysons ce que nous considérons comme deux points clefs : la modélisation des requêtes et des plans de traitement d'images.
  - Un modèle de requête : dans BORG, l'accent a été mis sur la définition d'une grammaire formelle dédiée à la formulation des requêtes de traitement d'images. Une requête correspond à une tâche à effectuer sur une image, dans un contexte donné et suivant un certain nombre de contraintes. Pour concrétiser ces concepts, la grammaire décrite sur la Figure 3.14 est utilisée.

```

Request ::= ((<task> <constraint> *) <context> <image>)
Task ::= (<goal><object-type> <specific-feature>*)
Specific-feature ::= <attribute> <value>
Constraint ::= (<attribute> <relation> <value>)
Context ::= ((<attribute> = <value>))
Goal ::= enhance | extract | isolate | segment .....
Object-type ::= region | bundary | background | contour | zone | object ...
Attribute ::= region-boundary | object-color | camera-type | contour-type ...
Relation ::= < > | = | <> ...
Value ::= <numeric> | <symbolic>
Symbolic ::= high | big | small | precise | sharp
Image ::= « filename »
    
```

*Figure 3.14 : Les modèles de requêtes dans Borg*

- Un modèle de plan de traitement d'images : du point de vue planification, les plans de traitement d'images sont représentés dans un arbre de tâches ET / PUIS suivant cinq niveaux d'abstraction : requêtes, objectifs, fonctionnalités, procédures et opérateurs. Cette hiérarchie ne traduit pas une décomposition fonctionnelle mais une décomposition d'affinements. Chacune des tâches représentées par les nœuds de l'arbre partage de ce fait la même liste d'attributs : *But*, *Contraintes*,

*Contexte, Etat, Entrées, Décomposition, Chemin, Sorties, Evaluation, Résultat, Poids.* Parmi ces attributs figurent les différents « mots » de la grammaire de définition des requêtes. Une requête est en effet une tâche du niveau le plus élevé dans l'arbre. De plus, les attributs entrées et sorties correspondent aux flots de données, soit entre une tâche et ses filles, soit entre tâches sœurs.

- Les connaissances opératoires : le dernier niveau de l'arbre de tâches concerne les connaissances opératoires. Il est pourvu d'attributs supplémentaires caractéristiques de ce type de connaissances (cf. II.3.1.). En effet, si une condition préalable à la construction d'un système de traitement d'images est de disposer d'une librairie de traitement d'images à la Khoros [Rasure 1994], regrouper quantité d'opérateurs sous forme d'exécutables dans une librairie n'est pas suffisant. Pour avoir une chaîne efficace, s'adaptant à une image donnée, il est indéniablement nécessaire de modéliser ces opérateurs pour en ajuster les paramètres et en contrôler les itérations. Beaucoup d'efforts ont été faits dans le système BORG à ce niveau. Parmi les points innovants que celui-ci propose, l'attribut concernant le mode d'exécution mérite d'être souligné. Trois modes sont inclus : le mode normal, qui correspond à une exécution unique de l'opérateur ; le mode boucle (*loop mode*), qui correspond à plusieurs exécutions successives du même opérateur ; et enfin le mode optimisation (*optimization mode*) qui consiste à faire varier les paramètres utilisés dans un intervalle fixé jusqu'à satisfaction de contraintes données. Hormis ce mode d'exécution, les autres attributs utilisés sont plus classiques, il s'agit du coût en temps ainsi que du prototypage de l'opérateur.

Nous verrons par la suite que nous nous sommes inspirés de ce travail, à la fois pour l'expression de requêtes, mais aussi pour l'aspect représentation de stratégies par un arbre de tâches.

### *iii. Le système BBI de Lefèvre [Lefèvre 1994]*

**BlackBoard Image (BBI)** est un système d'aide à la photo interprétation d'images aériennes issues de différents capteurs (infrarouges, radar, visible...). Il a été développé sur une architecture multi-processeurs à base de transputers. Dans ce système, relativement proche du nôtre à de nombreux points de vue, un réel effort a été entrepris au niveau représentation des connaissances mises en jeu aussi bien du point de vue descriptif, qu'opératoire et stratégique.

- Les connaissances descriptives : quatre classes d'objets à reconnaître sont modélisées dans BBI : le milieu naturel, les voies de communication, les bâtiments et les véhicules. Pour chacune de ces classes, neuf critères de description sont définis : la couleur, le contraste, l'homogénéité, la texture (niveau image), la forme, la géométrie, la taille (niveau « intermédiaire »), la localisation et le positionnement (niveau symbolique). Pour chacun de ces critères, une valeur symbolique est définie de façon relative en fonction du contexte.

- Les connaissances opératoires : une partie des connaissances opératoires est également explicitée dans ce système. Chacun des traitements utilisés dans le système est décrit de la façon suivante :
  - ❑ La description de l'algorithme proprement dit.
  - ❑ La nature des informations en entrées / sorties.
  - ❑ Le type des entrées, des sorties, et des paramètres de contrôle.
  - ❑ Un ensemble de caractéristiques « techniques » : la parallélisation envisageable, la complexité algorithmique, la taille en mémoire du code exécutable ...
- Les connaissances stratégiques : la représentation des connaissances stratégiques dérive directement de celle des connaissances opératoires décrites ci-dessus. En effet, à partir de la définition des traitements disponibles dans la bibliothèque, le système est capable de générer un graphe général de stratégie de détection associant les traitements adéquats. Ces traitements sont implémentés sous formes d'agents qui peuvent négocier en utilisant le protocole des réseaux contractuels. Ce mécanisme consiste à sélectionner les opérateurs les plus adaptés au contexte traité. Pour cela les agents évaluent leur compétence à résoudre le problème qui leur est confié, en exploitant dans ce cadre la théorie de Demspster Shafer.

Les neuf critères définis ci-avant sont également utilisés pour l'élaboration de la stratégie. En effet, des tables de robustesse de critères de détection en fonction du contexte ont été définies. Elles permettent de déterminer, pour l'extraction de chaque objet, la pertinence d'utilisation des différents critères. Une échelle a été définie pour caractériser cette pertinence. Elle va de (-) qui correspond à un critère ne permettant pas la caractérisation d'une zone à (+++) qui correspond à un critère indispensable. Les tables ainsi constituées font partie des connaissances stratégiques du système. En utilisant ce mécanisme, les chaînes de traitement associées aux critères principaux de détection (+++) sont dans un premier temps utilisées avant, si nécessaire, de lancer les chaînes moins « sûres » dans le cas d'un échec des précédentes.

Nous verrons que la distribution des connaissances sous forme d'agents ainsi que le protocole de négociation utilisé dans BBI ont inspiré nos travaux.

### **3.3.3. Un bilan global des approches de la littérature**

Dans cette section, nous dressons un bilan global des approches adoptées du point de vue ingénierie des connaissances par les systèmes décrits dans la littérature.

Concernant les connaissances descriptives, les différents choix pour la représentation des objets présents sur le document ou sur la scène sont relativement proches. La notion de « réseau hiérarchique

d'objets » au sens large semble s'imposer, que le formalisme adopté soit les *frames*, les réseaux sémantiques ou les systèmes multi-agents. L'intégration des relations de décomposition, de spécialisation et de topologie semble incontournable. Toutefois, ces notions sont rarement toutes explicitées dans les systèmes étudiés. De plus, l'étape d'acquisition des connaissances, visant à initialiser les systèmes, est très souvent oubliée. La généralité des systèmes souffre de cette lacune, et le passage d'une application à une autre exige toujours des retours importants sur le code du système. Citons toutefois les travaux de [Pasternak 1993] et [Ah Soon 1998] pour le document et de [McKeown 1989] pour les systèmes de vision, qui proposent des solutions partielles dans ce cadre, pour des problèmes particuliers. Outre cette acquisition, l'évolution des connaissances descriptives au cours de la vie du système est également rarement évoquée dans l'ensemble des systèmes. Les modèles d'objets, quels qu'ils soient, sont statiques, et n'évoluent pas au cours de la vie du système.

Concernant l'aspect opératoire, force est de constater que les systèmes d'interprétation de documents techniques ont jusqu'alors délaissé cette problématique. Dans les systèmes de vision, l'approche du système BORG est très intéressante. Elle innove en définissant, en plus des attributs classiques définis par exemple dans BBI (définition des entrées /sortie, des paramètres...), un « mode d'exécution » qui permet à l'ordonnanceur de tâches d'appliquer un opérateur donné de façon itérative en fonction de contraintes sur les résultats. Néanmoins, hormis ce système, il peut être constaté que les équipes travaillant à l'adéquation contexte/traitement sont encore trop rares dans le domaine.

En terme de connaissances stratégiques, beaucoup de systèmes proposent des approches intéressantes. La notion de cycle perceptif, présente dans [Ogier 2000a] et [Ramel 1998] semble appropriée pour assurer la robustesse des résultats. Les notions de coopération et de compétition, souvent présentes dans les systèmes d'analyse de scènes permettent également d'assurer une fiabilité accrue des résultats. Toutefois, aucun système d'interprétation de documents techniques n'exploite de telles approches. Cette avance des systèmes de vision sur les systèmes d'interprétation de documents techniques se retrouve également au niveau de la définition des objectifs du système. En effet, la plupart des systèmes d'analyse de documents techniques voient le problème de l'interprétation comme un processus s'appliquant sur l'intégralité du document, visant à tout extraire. Il n'y a pas eu vraiment de réflexion concernant l'expression d'objectifs du système.

Dans le chapitre qui suit, la première version de notre système NATALI V1 est présentée. Nous montrerons que celle-ci apporte quelques solutions aux problèmes évoqués dans ce bilan mais qu'elle est encore sujette à de nombreuses lacunes. Nous proposerons ensuite de premières pistes pour remédier à celles-ci dans le chapitre qui suivra.

### **3.4. L'expérience de *NATALI v1***

Dans le cadre du projet initial *NATALI*, un premier système opérationnel d'interprétation de documents techniques a été conçu. Les objectifs de ce système (nommé *NATALI v1* par la suite) étaient de numériser massivement la documentation réseau France Telecom, afin d'alimenter la montée en charge d'applications de gestion de ce réseau (l'application 103 France Telecom). Ce système, issu d'un travail d'équipe initié en 1998, a été implémenté et son utilisation nous a permis d'obtenir et d'analyser de premiers cas d'usage. Dans ce chapitre, après avoir défini les objectifs de ce travail, nous dressons une synthèse de l'approche retenue. Cette synthèse est orientée ingénierie des connaissances, conformément à l'esprit de cette seconde partie. Elle est complémentaire à la présentation davantage orientée génie logiciel réalisée dans [Grenier 2000, 2001]. L'intégration logicielle de *NATALI v1* est ensuite brièvement décrite. Puis, une analyse critique basée sur les conclusions de l'étude bibliographique et sur de premiers retours d'expérience est dressée.

#### **3.4.1. Les objectifs**

Confronté à une forte diversité de représentation des plans de réseau France Telecom, l'objectif premier du projet *NATALI* était de pouvoir généraliser le système développé aux différentes normalisations de documents au sein de l'entreprise. Le système devait ainsi pouvoir s'adapter, sans modifications majeures, à différentes variantes de représentation du document. Pour satisfaire ce but de généralité, les principes suivants avaient été définis :

- les connaissances utilisées devaient être au maximum externalisées du système de traitement, en évitant de les ancrer dans le code source ;
- le système devait gérer le plus dynamiquement possible l'enchaînement des traitements, en fonction de la bibliothèque de traitements disponibles et du contexte à un instant donné ;
- il devait être facile d'ajouter de nouveaux traitements, en vue de traiter le cas échéant un nouveau type de problème ;
- le système devait être le plus autonome possible en ayant la capacité de vérifier l'adéquation des données qu'il produit avec le modèle. En cas de non-adéquation, et s'il ne disposait pas de moyen de corriger les données erronées, il devait pouvoir alerter l'utilisateur.

Quiconque s'est intéressé au problème de l'interprétation d'un document technique sait que concevoir un système répondant à l'ensemble de ces contraintes constitue un travail de longue haleine. Les récentes habilitations à diriger des recherches soutenues dans le domaine [Tombre 1996], [Mullot 1999], [Ogier 2000b], le GDR associé à la thématique [Garbay 1998], ainsi que les travaux présentés et les discussions lors des récentes conférences GREC'01 et ICDAR'01 en témoignent. Dans la suite

de ce chapitre, nous exposons les principales idées retenues par notre équipe pour la réalisation de la première version du système, en distinguant les parties implémentées des parties restées au niveau conceptuel. Nous adoptons, pour ce bilan, l'articulation utilisée lors de l'étude bibliographique précédente en décrivant le système à travers les choix effectués pour modéliser, puis représenter les connaissances descriptives, opératoires et stratégiques.

### 3.4.2. Les connaissances et leur représentation

#### 3.4.2.1. Les connaissances descriptives dans NATALI v1

Concernant les connaissances descriptives, l'objectif principal dans NATALI v1 était d'identifier l'ensemble des données manipulées par le système pour répondre à une requête d'interprétation émise par un utilisateur. Dans ce cadre, la scission classique entre données génériques et données spécifiques a été adoptée. La Figure 3.15 illustre cette scission. La partie dite générique (partie basse) concerne les objets du traitement d'images et de la reconnaissance de formes : vecteurs, arcs (non implémenté), caractères, occlusions, formes connexes. Le qualificatif de générique vient du fait qu'elle concerne des objets rencontrés quasi-systématiquement sur les documents techniques. La partie dite spécifique (partie haute) concerne quant à elle les objets relatifs aux connaissances du domaine, qui varient donc d'un type de document à un autre. Les liens qui sont représentés sur cette figure sont des relations de constitution. En effet, un objet, qu'il soit du domaine ou utilisé en traitement d'image, est toujours décrit par une liste de constituants. Cette relation de constitution apporte évidemment une structure fortement hiérarchique au modèle, d'où la notion de « *modèle en couches* » [Labiche 1998] qui sera fréquemment évoquée par la suite.

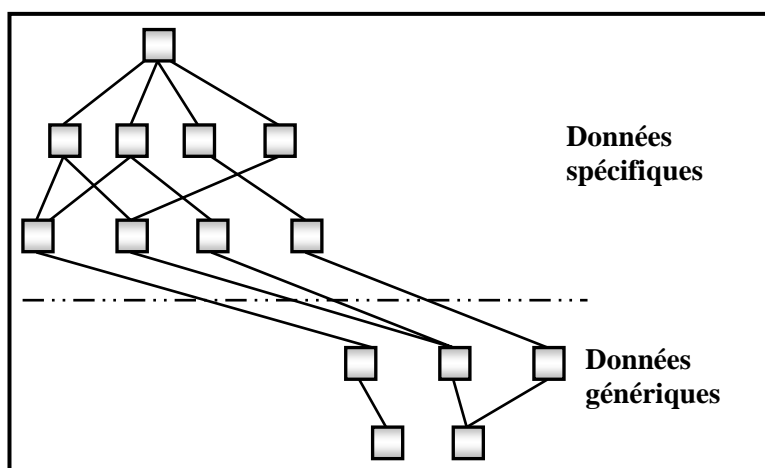


Figure 3.15 : Scission spécifique / générique du modèle de connaissances descriptives

Dans le système NATALI v1, dédié à l'interprétation d'une partie des plans de réseau France Telecom, les objets suivants au niveau du domaine (spécifique) ont été identifiés :

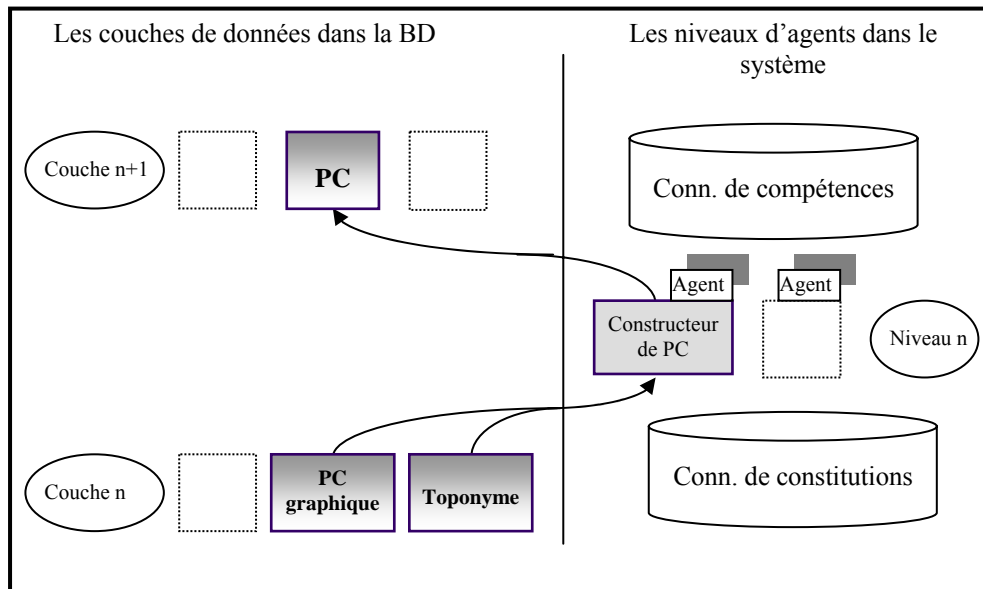
- *Réseau* qui regroupe l'ensemble des équipements du réseau
- **TAR** : (Tronçon d'**AR**tère, *i.e.* les câbles), **PC** (**P**oint de **C**oncentration), *chambre*.
- *toponymes*, *TAR graphique*, *chambre graphique*, *PC graphique*. Dans les trois derniers cas, il s'agit des représentations symboliques des équipements *TAR*, *chambre* et *PC*.

A partir de l'identification de cet ensemble d'objets et des relations de constitutions, il s'agissait ensuite, tant au niveau du domaine qu'au niveau traitement d'images et reconnaissance de formes, d'intégrer ces connaissances de façon à pouvoir opérationnaliser le système. Pour ce faire, compte tenu de la quantité et de la diversité des connaissances mises en jeu lors d'un processus d'interprétation, nous avons opté pour une distribution de celles-ci. Le paradigme multi-agent a donc été choisi. Dans ce contexte, les connaissances sont représentées suivant trois formalismes :

- Les agents intégrés dans un **Système Multi-Agents (S.M.A.)**. Puisque l'objectif de tout système d'interprétation de documents est la recherche des données présentes sur le document, celles-ci doivent être associées à une phase de « construction ». Dans *NATALI v1*, ce sont des agents qui permettent une telle construction. Chacun des objets définis comme potentiellement présents sur le document est lié à un agent qui permet d'en extraire les instances. Pour ce faire, les agents disposent d'une liste de sous-éléments (répertoriés dans un fichier de configuration), qui sont les données nécessaires à la construction. Avec cette connaissance des liens de composition, et grâce à ses compétences de communication, l'agent peut ainsi déléguer des tâches à d'autres agents. Etant donnée la structure fortement hiérarchique du réseau, due à l'omniprésence de la relation de composition, les agents sont répartis en niveaux et seuls communiquent les agents de niveaux adjacents. Des connaissances descriptives sont donc implicitement intégrées dans la répartition en niveaux des agents.
- Le code des constructeurs de données expriment donc également une partie de la connaissance descriptive liée aux relations intra-couches. En effet, les connaissances correspondant aux relations entre objets de la même couche sont intégrées dans les constructeurs de données des agents, de façon procédurale. Typiquement, les connaissances sur le voisinage sont représentées de cette façon.
- Le schéma de la base de données relationnelle. Pour le stockage des résultats courants de l'interprétation, notre choix s'est porté sur l'utilisation d'une base de données relationnelle. Celle-ci contient donc des structures de données correspondant à chacun des objets identifiés. Chacune de ces données est munie d'attributs qui contiennent les propriétés des objets. Les relations de



composition sont intégrées dans le schéma de la base de données sous la forme d'attributs particuliers permettant de décrire les liens de composition.



*Figure 3.16 : L'opérationnalisation des connaissances descriptives dans NATALI v1*

La Figure 3.16 illustre la représentation des connaissances descriptives dans le cas d'équipements particuliers appelés **Points de Concentration** (cf. Annexe B). Ces équipements sont représentés par un symbole (le PC graphique), auquel est associée une chaîne de caractères (toponyme). Dans le système *NATALI v1*, un objet *PC* est représenté par :

- L'agent *PC*, et sa position dans la structure du SMA. Cet agent doit être placé sur la couche supérieure à celle contenant les agents « *Toponyme* » et « *PC graphique* » (partie droite de la Figure 3.16 et Figure 3.17).
- Un fichier texte, contenant l'information de composition, *i.e.* « *TOPONYMES* » ET « *PC\_GRAPHIQUE* » (noté « Conn. De constitution » sur la figure)
- Une structure de données contenant les différentes propriétés de l'objet dans la base de données (partie gauche de la Figure 3.16)
- Le code du constructeur intégré dans l'agent *PC* qui implémente le fait que la relation utilisée pour cette construction est le voisinage.

Outre cet agent *PC*, 12 autres agents ont été intégrés dans *NATALI v1*, la Figure 3.17 montre la répartition de ces agents (un agent par type de donnée) sur les 6 couches du système, ainsi que les liens de constitutions entre les données correspondantes.

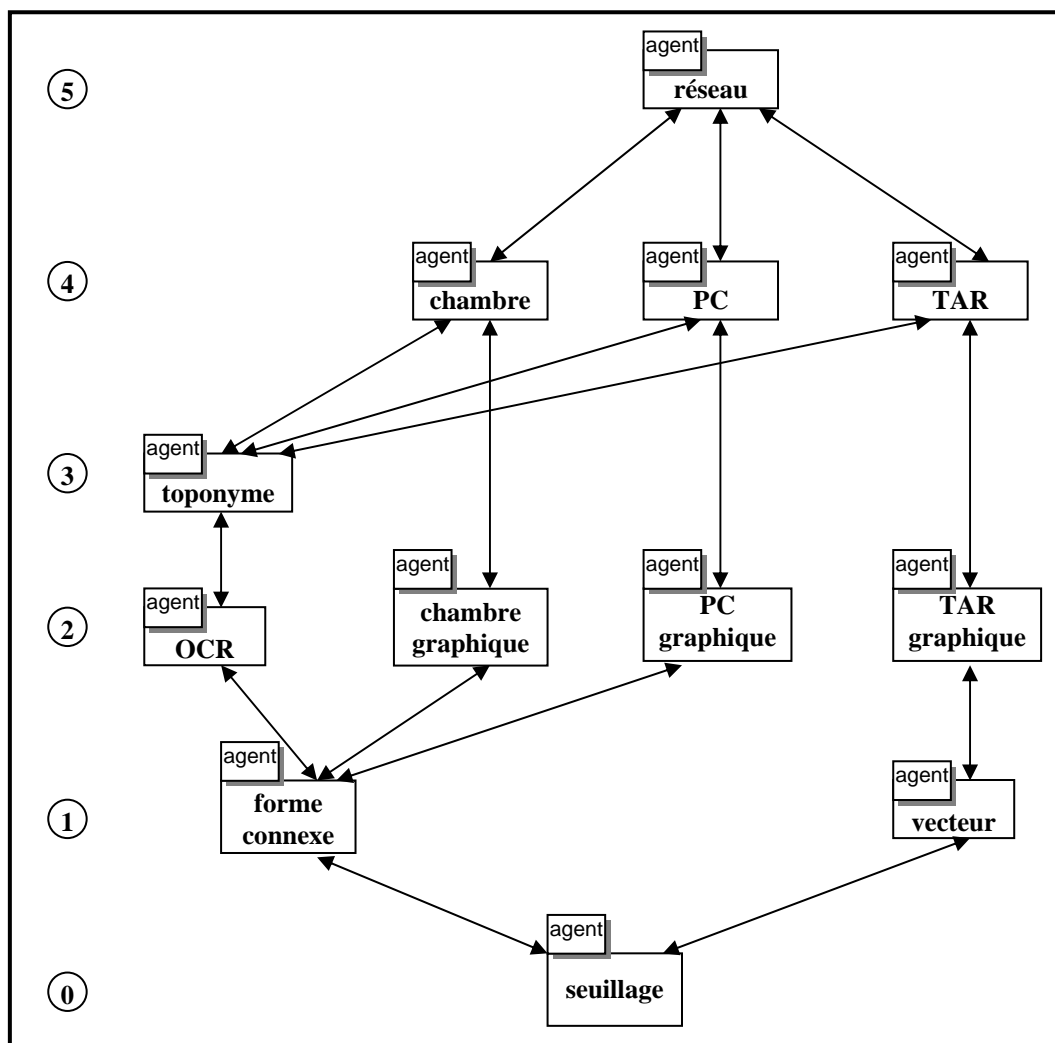


Figure 3.17 : Les agents implémentés dans NATALI v1

### 3.4.2.2. Les connaissances opératoires dans NATALI v1

Dans la première version de *NATALI*, aucune connaissance concernant les différents traitements implémentés dans la chaîne d'interprétation n'a été intégrée sous forme explicite. Les différents outils utilisés pour l'interprétation (qui sont intégrés dans les agents décrits ci-dessus) sont issus d'une bibliothèque de traitements d'images et de reconnaissance de formes du laboratoire. Ces outils sont documentés mais cette documentation n'intègre que les entrées / sorties des algorithmes. Les paramètres des algorithmes ainsi que leurs lois de réglage en fonction de l'intention et du contexte courant ne sont pas pris en considération dans cette documentation. Sur ce point, qui constitue actuellement une préoccupation importante de la communauté ([Ogier 2001], [Gardes 2001], [Pridmore 2001]...), des travaux concernant l'extraction de connaissances d'outils de traitement d'images sont en cours au laboratoire PSI [Saidali 2001]. Ces travaux devraient permettre, à terme, d'intégrer dans le système l'expertise du traiteur d'images afin d'ajuster les paramètres de traitement

en fonction du contexte et de l'objectif. Nous reviendrons sur ce point en proposant quelques axes possibles dans la suite du mémoire.

### **3.4.2.3. Les connaissances stratégiques dans *NATALI VI***

Dans le système *NATALI VI*, un gros effort a été consenti pour que le système puisse utiliser différentes stratégies pour l'interprétation des documents. Le système fonctionne à partir de deux informations fondamentales : le contexte courant de l'interprétation à un instant  $t$  (*i.e.* les données déjà construites) ainsi que l'intention de son utilisateur. Deux principales originalités mises en œuvre pour ce faire doivent être ici soulignées :

- Le contrôle d'exécution des agents. Le système étant distribué, c'est le contrôle de chacun des agents qui détermine la stratégie globale du système. Ce contrôle des agents est basé sur le protocole des réseaux contractuels que nous présentons ci-après en (i.).
- Le formalisme d'intention, représenté par l'intermédiaire de requêtes qui se *propagent* d'abord, lors d'une phase descendante, dans les couches du système et qui *dirigent* la reconstruction progressive ascendante des informations jusqu'au niveau demandeur. Ce formalisme est exposé en (ii.)

#### ***i. Le contrôle dans *NATALI VI****

Nous avons vu ci-avant que les connaissances descriptives du système avaient été implicitement représentées par un système multi-agents hiérarchique. Ce système est initialisé par une intention qui est exprimée par l'utilisateur et qui respecte un formalisme qui sera décrit en (ii.). Le fonctionnement du système est ensuite basé sur la « propagation / délégation » de la tâche exprimée par cette intention. Cette tâche est en effet décomposée en sous tâches dans les différents niveaux du système. Cette délégation est effectuée en utilisant le protocole des réseaux contractuels, comme dans [Lefèvre 1994]. Un tel protocole consiste en 4 étapes :

- (1) Appel d'offres : un agent demande un service aux agents du niveau inférieur. Dans le contexte dans lequel nous sommes, ce service correspond à la construction des données.
- (2) Envoi de propositions : Les agents ayant reçu le message répondent, positivement ou négativement.
- (3) Attribution de contrat(s) : l'agent demandeur choisit le(les) agent(s) avec lequel(lesquels) il va coopérer.
- (4) Acquiescement des contrats : l'(les) agent(s) sélectionné(s) effectue(nt) la tâche demandée, et transmet(tent) le résultat de leur travail à l'agent demandeur.

Au sein des agents, c'est une base de règles (appelée scénario interne) qui gère l'ensemble de cette stratégie. Après réception d'une requête de construction (ii), l'agent va consulter la base de règles qui lui est propre, et il va déclencher les règles qui peuvent l'être en fonction du type de message arrivant. Pour cela, une grammaire a été définie et un analyseur syntaxique traite les messages reçus, afin de consulter la base de données, de la modifier, de sous traiter... Nous présentons ce langage ci-après.

Notons que dans certains cas, une évaluation de la cohérence des données construites est effectuée. Si celle-ci n'est pas jugée satisfaisante, il était initialement prévu que l'agent qui détecte l'incohérence puisse relancer de nouvelles requêtes sur d'autres agents. Néanmoins, nous n'avons pas implémenté cette fonctionnalité dans *NATALI v1*.

D'un point de vue global, une stratégie d'interprétation est donc implicitement un arbre dans lequel chaque nœud prend une décision quant aux fils qu'il va activer, par l'intermédiaire d'une base de règles. Dans la première version du système, un unique agent est associé à chaque type de donnée symbolique. Nous n'avons pas mis en œuvre de compétition entre constructeurs par manque d'outils, les réponses de compétence aux requêtes sont donc binaires et de ce fait la négociation n'existe pas.

## *ii. Les requêtes dans NATALI VI*

L'une des bases fondamentales du paradigme multi-agents est la communication entre agents [Ferber 1997a, 1997b]. Dans notre système, ces communications sont utilisées pour la mise en œuvre du protocole des réseaux contractuels et pour la communication avec l'utilisateur. Dans ce but, un formalisme de requêtes a été mis en place. Celui-ci est basé sur le langage **KQML** (**K**nowledge **Q**uery and **M**anipulation **L**anguage), issu des travaux du consortium **KSE** (**K**nowledge **S**haring **E**ffort) pour la définition de spécifications. L'idée de ce langage est de permettre le partage et la réutilisation de bases de connaissances et de systèmes à base de connaissances. Il est basé sur la théorie des actes de langage, le lecteur en trouvera une excellente description dans [Finin 1993]. Un message KQML, est composé d'un acte de communication performatif et d'une série de paramètres variant selon le type d'acte de communication. Dans ce cadre, une requête (un message) est représentée par une chaîne de caractères devant respecter une grammaire fournie par la spécification KQML. Dans cette grammaire, les paramètres sont représentés par des paires : « :clé valeur ». Dans chacune d'elles, la clé est un mot prédéfini et la valeur peut être un mot ou une expression (ensemble de mots entre parenthèses). La norme prévoit un ensemble de paramètres prédéfinis, mais il est tout à fait possible de créer de nouveaux paramètres. Les paramètres que nous utilisons sont explicités dans [Grenier 2001]. Le langage ainsi défini est le langage qu'utilisent les différents agents du système pour communiquer entre eux. De plus, l'utilisateur pouvant être considéré comme un agent particulier, il utilise ce même langage pour décrire son intention, certains des champs n'étant pas exploités. Le système est muni d'une interface permettant de générer la requête. La grammaire ainsi que les éléments terminaux et

non terminaux qui y sont associés constituent donc la représentation des connaissances concernant les requêtes qu'il est possible d'émettre. Nous verrons en 3.4.4 les avantages mais aussi les limites de ce langage.

Notons qu'avec une telle approche, l'utilisateur peut définir des intentions autres que celle consistant à interpréter tout le document. Le système étant distribué et fonctionnant par envoi de message, il est possible d'émettre une intention spécifique d'interprétation concernant n'importe quel objet. L'utilisateur peut ainsi choisir de rechercher sur le document toutes les occurrences d'un symbole particulier. Cela permet d'envisager l'utilisation du système en vue d'indexation de documents, ce qui constitue actuellement un axe stratégique de recherche pour France Telecom, mais aussi dans toute notre communauté.

### **3.4.3. L'implémentation de *NATALI v1***

En terme d'implémentation, la conception de ce premier système opérationnel est de type multi-thread, multi-agents et multi-plateformes. Il exploite la complémentarité de deux langages de programmation, le langage objet JAVA pour la programmation des agents et des interfaces, et les langages C/C++ pour les traitements d'images. Les communications entre agents s'effectuent par l'intermédiaire d'un bus logiciel implémentant la norme CORBA (en utilisant le service des canaux d'événements). Cette norme est également utilisée pour l'appel de méthodes distantes, grâce au service de nommage. En effet, les traitements d'images sont exécutés sur une machine IRIX alors que les agents sont implémentés sur un PC. Les données sont stockées au sein d'une base de données Oracle 8, dont le serveur est implémenté sur un PC. Ces points sont illustrés Figure 3.18 et Figure 3.19. Ces choix ont permis de développer un système présentant du point de vue logiciel de nombreux avantages, qui rejoignent les préoccupations actuelles de la communauté génie logiciel :

- La portabilité qui devient un sujet prédominant pour la conception de composants réutilisables, et qui s'avère indispensable dans le cadre de collaborations [Lamiroy 2001].
- La modularité : les agents sont construits par instanciation d'une liste de modules indépendants, il est donc possible de modifier l'un des modules sans trop affecter les autres.
- La réutilisabilité : grâce à la modularité, il est possible de réutiliser certains des modules des agents, comme le module de communication par exemple. Cela permet à un spécialiste de génie logiciel d'intégrer assez facilement de nouveaux agents.

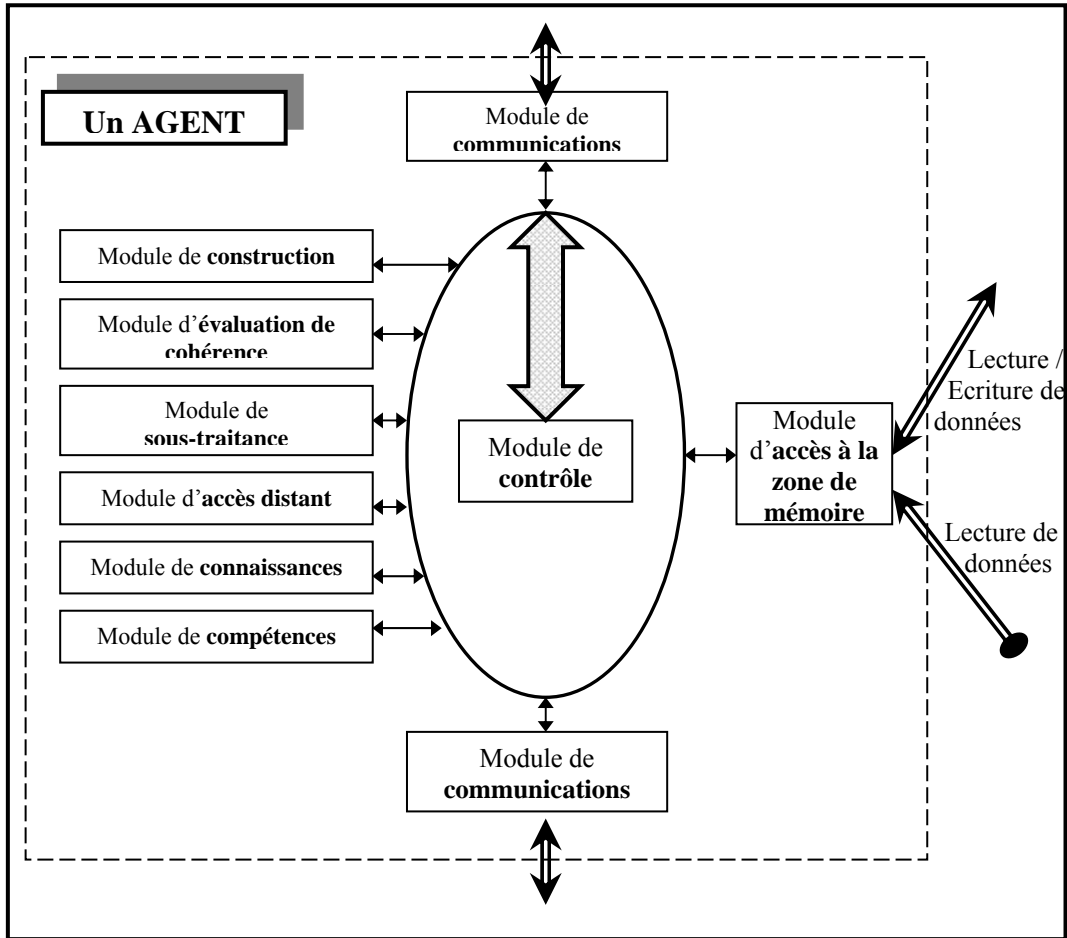


Figure 3.18 : Les modules d'un agent dans NATALI v1

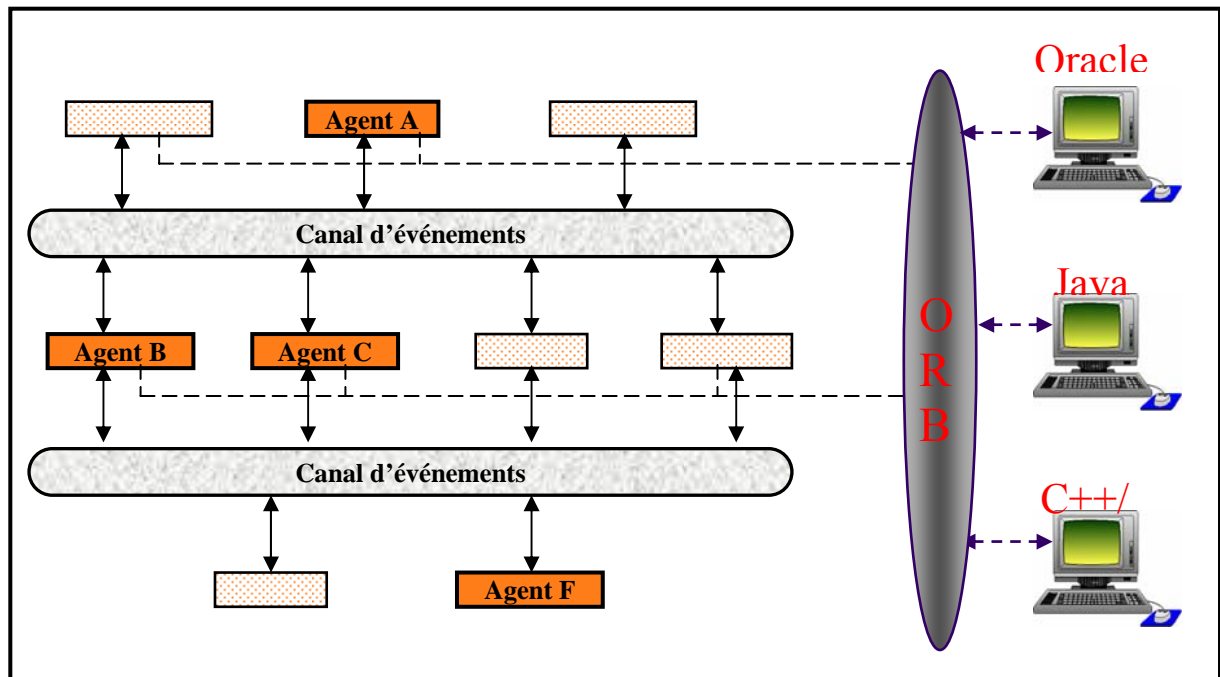


Figure 3.19 : L'architecture logicielle de NATALI v1

Après cette description succincte du système *NATALI v1*, du point de vue « Ingénierie des connaissances », puis du point de vue implémentation, nous proposons ci-après une analyse critique de celui-ci. Cette critique, issue de premiers cas d'usage, nous permettra de mettre en avant les points forts et les lacunes du système afin de présenter la poursuite de nos travaux dans le cadre de *NATALI v2*, mais aussi pour les travaux ultérieurs de notre équipe.

#### **3.4.4. Analyse critique**

Pour la conception de ce premier système, une réflexion a été menée sur l'ensemble des objets manipulés lors d'un processus d'interprétation des documents « réseau » France Telecom. Une identification de la partie générique (indépendante du domaine) et de la partie spécifique (dépendant du domaine) de ces données a été effectuée. Cette distinction, assez classique, est néanmoins très intéressante puisqu'elle permet de développer une partie du système sans se soucier du domaine considéré par le document. Cette partie restera de ce fait valable quel que soit le document. En outre, l'intégration de connaissances du domaine dans un système d'interprétation de documents techniques reste rare, comme nous l'avons montré dans l'étude bibliographique. Ce point offre, nous le verrons ci-après, des perspectives très intéressantes. Bien que cette distinction générique/spécifique soit réalisée, les deux catégories de connaissances identifiées sont représentées dans un formalisme unifié : un système multi-agents associé à une base de données. Or, les connaissances d'un spécialiste de reconnaissance de formes sont d'un tout autre ordre que celles d'un expert du domaine ; les représenter dans le même formalisme n'est donc pas nécessairement adéquat. La notion de cycle de gestion de la cohérence rejoint cette remarque, puisque c'est au niveau des connaissances du domaine que la cohérence d'une donnée pourra être déterminée. Il n'y a selon nous pas de notion de cohérence parmi les données génériques, tant qu'un modèle métier n'est pas utilisé. Il n'est par exemple pas possible de juger de la cohérence d'un vecteur reconstruit tant que le « sens » de ce vecteur n'est pas intégré.

Au niveau métier, la plus grande partie des connaissances est représentée procéduralement, seules les relations de constitution sont explicitées. La relation de voisinage, présente dans les différents exemples qui ont précédé, constitue une illustration des connaissances représentées procéduralement. De plus, comme nous l'avons évoqué en 3.2.2.1, les relations entre concepts métier peuvent être beaucoup plus variées que celles de composition. Par exemple, il semble opportun d'y ajouter la relation de spécialisation (*sorte-de*), omniprésente dans les systèmes évoqués lors de l'étude bibliographique. A titre d'illustration, un système se voulant générique se doit selon nous d'intégrer la notion d'objet générique « symbole », qui reste valable quel que soit le type de document technique pris en compte. Nous reviendrons sur ce point en 3.5.2.1.

Au niveau des connaissances que nous avons appelées graphiques en 3.2.2.1 (les connaissances génériques sur la Figure 3.15), le choix unique des liens de constitution reliant ces concepts est également incomplet. Par exemple, la relation entre une forme et un vecteur de caractéristiques statistiques extraits de celle-ci n'est pas une relation de composition, mais plutôt une relation de description exprimant une modélisation de la forme dans un espace de représentation. Nous proposerons, en nous appuyant sur les travaux de la première grande partie du manuscrit, une évolution du système sur cet aspect « connaissance du spécialiste de reconnaissance de formes ». Cet aspect rejoindra un autre point qui a évolué par rapport à la première version du système et qui réside dans l'aspect acquisition, modélisation, représentation puis opérationnalisation des connaissances. En effet, l'étape d'acquisition des connaissances n'a pas été prise en compte dans *NATALI v1*. Un utilisateur du système d'interprétation (éventuellement un « naïf » pour le domaine du traitement d'images et de la reconnaissance de formes) doit, pour passer d'une normalisation à une autre, intégrer de nouveaux agents dans le système, et éventuellement développer un nouveau traitement d'images à intégrer dans celui-ci. Il n'y a pas de phase d'acquisition de connaissances qui pourrait consister en une saisie conviviale des connaissances qui pourraient ensuite être opérationnalisées en SMA. Nous proposerons sur ce point quelques premiers développements ainsi que des perspectives précises pour la poursuite de ces travaux.

En revanche, il semble important de souligner la pertinence de l'utilisation d'un langage de requête. Le fait que l'utilisateur dispose d'un moyen de communiquer avec chacun des agents du système par l'intermédiaire d'un langage formalisé (KQML) est important et assez rare dans la littérature. Cela permet de sortir de l'écueil classique des systèmes classiques de rétro conversion où l'objectif est souvent le « tout interprété ». Cela rejoint, comme nous l'avons déjà précisé, les problématiques d'indexation actuelles de France Telecom, mais aussi de la communauté du document toute entière.

Enfin, soulignons le fait que l'implémentation de ce système a été très soignée, surtout en terme de modularité et de réutilisabilité des briques, dans le but d'obtenir un système opérationnel et évolutif du point de vue génie logiciel.



## **3.5. De l'analyse critique à NATALI V2**

### **3.5.1. Introduction**

Dans le chapitre 3.4, nous avons montré qu'un moteur puissant d'interprétation avait été développé dans le cadre de *NATALI v1*, mais que de nombreuses lacunes subsistaient du point de vue de l'ingénierie des connaissances dans ce système. Dans ce chapitre, nous présentons nos contributions concernant cet axe dans le cadre de la version 2 du système. Ces contributions sont de deux principaux ordres :

- Concernant les connaissances du domaine (connaissances spécifiques), il s'est agi d'intégrer une représentation explicite des connaissances identifiées dans *NATALI V1*, afin d'aller dans le sens de la généralité du système. Nous avons choisi un formalisme à base de réseau sémantique, qui intègre les concepts manipulés au niveau du domaine, mais aussi les différentes relations possibles entre ces concepts. Concernant ce réseau, nous présentons d'abord le modèle objet qui a été mis en place, puis l'interface permettant l'acquisition des connaissances. Quelques premiers développements concernant la génération automatique du système multi-agent d'interprétation à partir de ce réseau sont alors évoqués. Ces différents points font l'objet de la section 3.5.2.
- Concernant les connaissances de traitement d'images et de reconnaissance de formes, un modèle de stratégie a été présenté en 2.1.3. Ce modèle a été opérationnalisé et de nouveaux agents, basés sur différents modèles de formes, ont été développés. Cela a permis de mettre en œuvre concrètement le protocole des réseaux contractuels avec des compétitions entre agents. Ce protocole étant basé sur des scores d'aptitude, une interface permettant, en outre l'initialisation (l'acquisition) de ces scores a été développée. Nous présenterons donc en 3.5.2 le modèle, l'interface d'acquisition des connaissances correspondantes ainsi que l'intégration de ces connaissances dans le système.

L'ensemble de ces contributions, ainsi que leur intégration dans nos travaux d'équipe sont représentés sur la figure qui suit. Sur la partie « acquisition » de cette figure apparaissent trois interfaces. Les deux premières, sur la gauche, ont été développées dans le cadre des projets *NATALI* et *DocMining*, elles sont respectivement détaillées en 3.5.2.2 et 3.5.3.2. Elles permettent l'alimentation de nos bases de connaissances (partie « modélisation/représentation » de la figure), dont la structure est également présentée dans ces sections. La partie basse de la figure correspond à l'étape d'opérationnalisation du système d'interprétation à partir des connaissances explicitées et l'évolution de ces bases de connaissances. Ces points sont détaillés dans la suite.

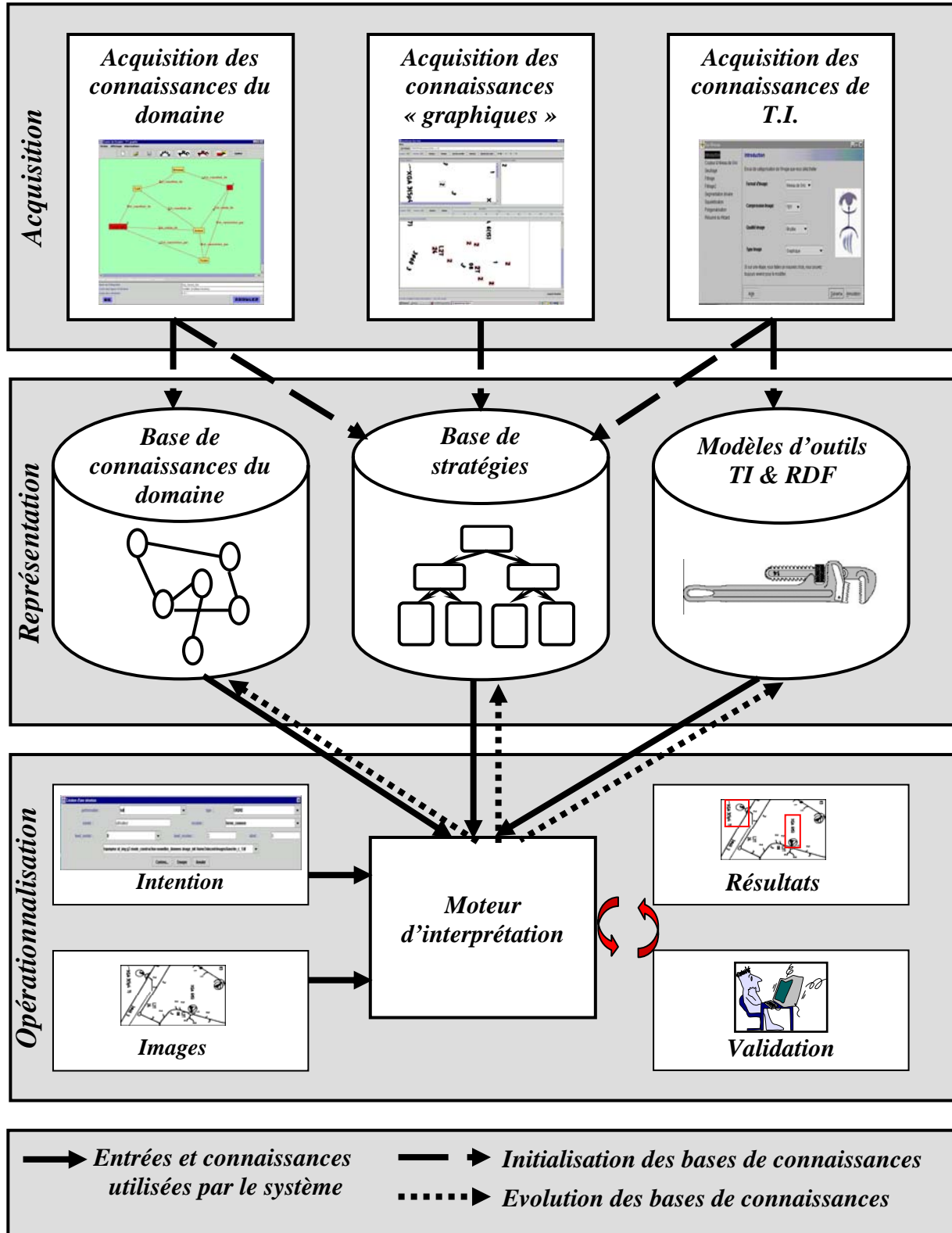


Figure 3.20 : Une vision du système d'interprétation du point de vue de l'ingénierie des connaissances

### **3.5.2. Les connaissances du domaine**

En 3.3, l'analyse de la littérature des systèmes d'interprétation de documents techniques a montré que peu d'équipes avaient abordé le problème d'une représentation explicite des connaissances du domaine. Ce point est très préjudiciable à l'évolution des systèmes dans la mesure où il contraint tout nouvel utilisateur à développer de nouveaux traitements en relations avec les données présentes sur les documents. En analysant plus largement les systèmes d'interprétation de signaux complexes, nous avons constaté que les travaux étaient plus avancés que dans ce domaine, et que de nombreux modèles développés en Intelligence Artificielle (règles, *frames*, réseaux sémantiques...) pouvaient être adoptés pour modéliser ces connaissances.

Dans *NATALI v1*, les concepts manipulés au niveau du domaine « Réseau France Telecom » ont été identifiés, au même titre que les liens de constitution entre ces concepts. Néanmoins, ces connaissances du domaine sont essentiellement représentées procéduralement (par le code des agents) dans le système développé (3.4.1). Un nouvel utilisateur de *NATALI v1* doit donc, pour définir les données présentes sur ses documents, intégrer de nouveaux agents construisant celles-ci. Ces agents doivent de plus être manuellement placés dans le système en fonction des liens de composition entre objets. Cette constatation se retrouve dans tous les systèmes analysés en 3.3.1. Aucun mécanisme d'acquisition de ces connaissances du domaine n'est présent. C'est pourquoi nous avons orienté nos réflexions sur ce point.

Dans ce chapitre, nous présentons les choix que nous avons effectués pour l'implémentation d'un formalisme explicite de représentation des connaissances du domaine. Ce choix est inspiré des réseaux sémantiques, et donc proche des travaux de [Den Hartog 1996], avec la notion de hiérarchie en plus. Il est basé sur une réflexion plus large, menée pour le développement d'une bibliothèque générique de gestion de graphes. Puis, nous présentons l'interface que nous proposons pour l'acquisition de ces connaissances du domaine. Enfin, nous montrons comment ce réseau est partiellement opérationnalisé par des agents.

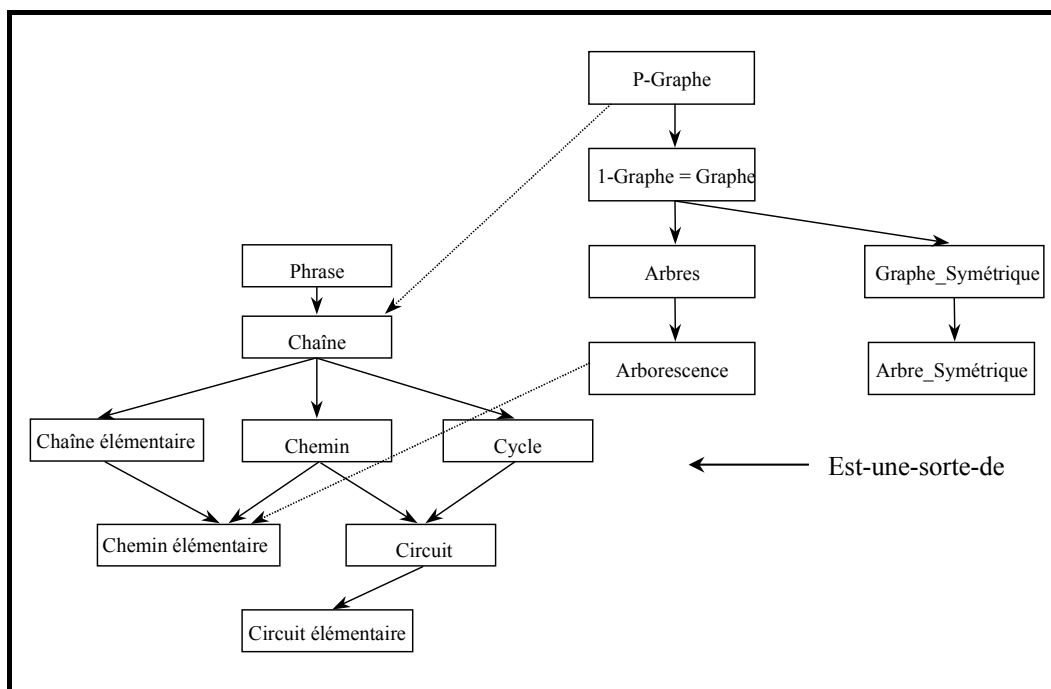
#### **3.5.2.1. La représentation des connaissances du domaine**

Pour représenter les connaissances du domaine, les formalismes à base de réseaux sémantiques constituent un formalisme intéressant. D'une part, la souplesse de leur structure permet de représenter n'importe quel type d'objet et de relation entre objets, et d'autre part, ils sont facilement compréhensibles grâce à leur aspect graphique. Nous avons donc conçu, dans le cadre de cette thèse, une bibliothèque permettant de stocker les connaissances du domaine pour l'interprétation sous formes de réseaux sémantiques. Ce travail s'est inséré dans le cadre d'une réflexion plus large, menée au sein du laboratoire pour le développement d'une bibliothèque de manipulation de graphes. En effet, les

graphes sont omniprésents dans notre domaine, et il était intéressant d'avoir un outil réutilisable. Dans cette partie, nous présentons donc d'abord succinctement cette bibliothèque du point de vue général, puis nous verrons l'implémentation particulière dans le cas des réseaux sémantiques, en l'illustrant avec le problème des documents réseaux France Telecom.

*i. Vers une bibliothèque générique de gestion de graphes*

Dans le domaine de la reconnaissance de formes et de l'interprétation, les graphes sont omniprésents. Les réseaux de Pétri, les réseaux de neurones, les schémas de Base de Données, les chaînes de Markov, les arbres de décision et encore bien d'autres structures sont tous des graphes particuliers. Nous avons donc réfléchi, dans le cadre de notre travail, à la conception d'une bibliothèque générique de gestion de telles structures. Partant des définitions de la théorie des graphes, nous en avons déduit le graphe d'héritage présenté en Figure 3.21 (l'ensemble des définitions et réflexions nous ayant amené à ce diagramme est disponible dans [Adam 1999b] au même titre que la spécification des classes).



*Figure 3.21 : Hiérarchie des classes dans la bibliothèque de gestion de graphes*

Sur un tel diagramme, les réseaux sémantiques se situent au plus haut niveau. En effet, plusieurs arcs peuvent joindre deux sommets puisque les relations entre deux concepts peuvent être diverses. Nous avons donc, dans le cadre de ce travail, utilisé la classe p-grappe dont les données membres et méthodes sont exposées dans [Adam 1999b]. Notons que cette bibliothèque est utilisée en Annexe B pour la reconnaissance des symboles, et que celle-ci est aussi exploitée dans [Heroux 2001] pour la reconnaissance de documents structurés.

ii. *La mise en œuvre du réseau sémantique pour NATALI v2*

Dans le cas de l'interprétation des documents réseau, une seule relation entre objets a été explicitée dans *NATALI v1*, celle de constitution (*est\_composé\_de*) et il n'y a pas de différence entre les différents types de données manipulés. Dans *NATALI v2*, nous proposons d'utiliser une catégorisation des objets du domaine en deux niveaux :

- Le premier niveau concerne la définition des objets du domaine. Ces objets ont pour propriétés importantes leur position géo-référencée, et le lien avec des bases de données existantes (d'autres sont envisageables). A ce niveau, deux principaux types de relations ont été identifiées : « sorte de » qui correspond en conception objet à la spécialisation des classes et « est\_constitué\_de » pour décrire les liens de constitution. La relation « est\_constitué\_de » permet, par exemple, de préciser qu'un tronçon d'artère est constitué d'une artère et de deux nœuds d'infrastructure. Les différents types de nœuds d'infrastructure sont ensuite définis par la relation « sorte-de ».

Dans le cas de l'interprétation de plans de réseau, cette partie des connaissances peut alors être représentée par le graphe suivant :

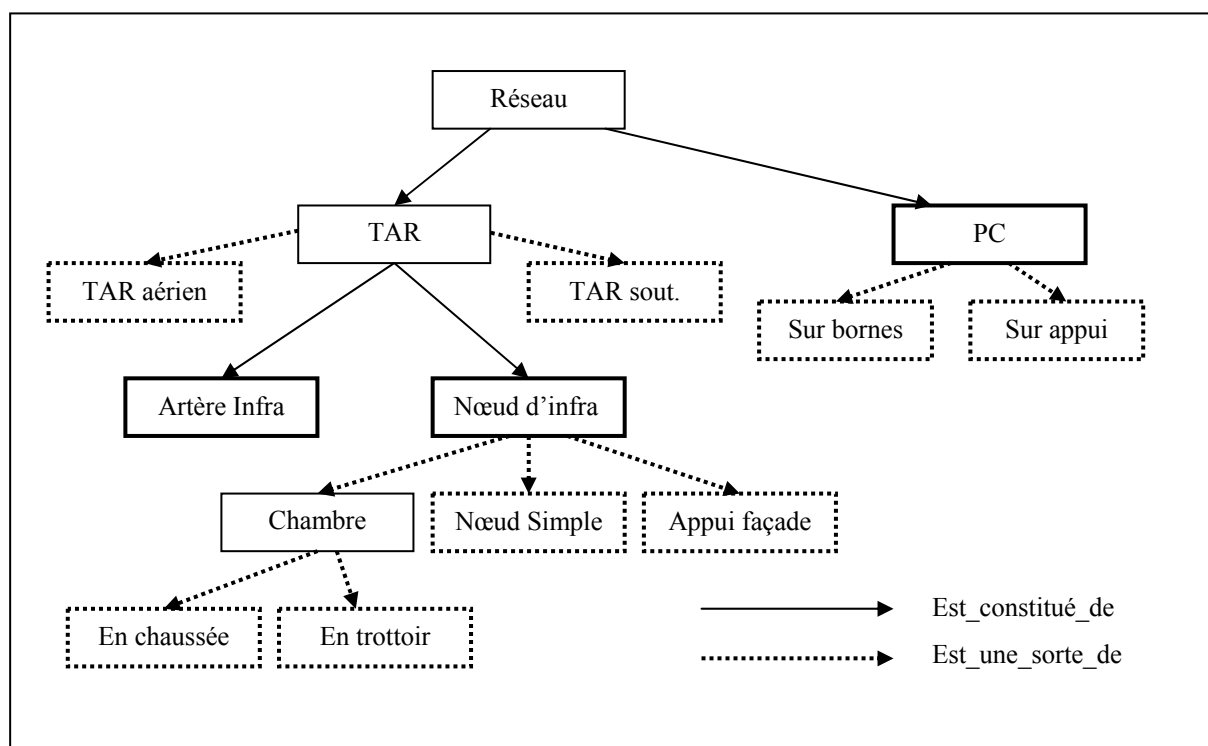


Figure 3.22 : La partie « définition des objets » pour les connaissances du domaine

Sur un tel réseau, nous nommons objets élémentaires tous les objets qui ne sont pas pères d'une relation de décomposition et qui ne peuvent donc pas être décrits par association de différents sous objets du domaine. Ils apparaissent en traits forts sur la Figure 3.22. (les PC, les artères et les Nœuds d'infra sur les documents France Telecom). Les objets qui apparaissent en traits pointillés sont quant à

eux issus d'une relation de spécialisation. Ce sont ces objets qui sont représentés sur les documents, ainsi que les objets élémentaires n'ayant pas de spécialisation.

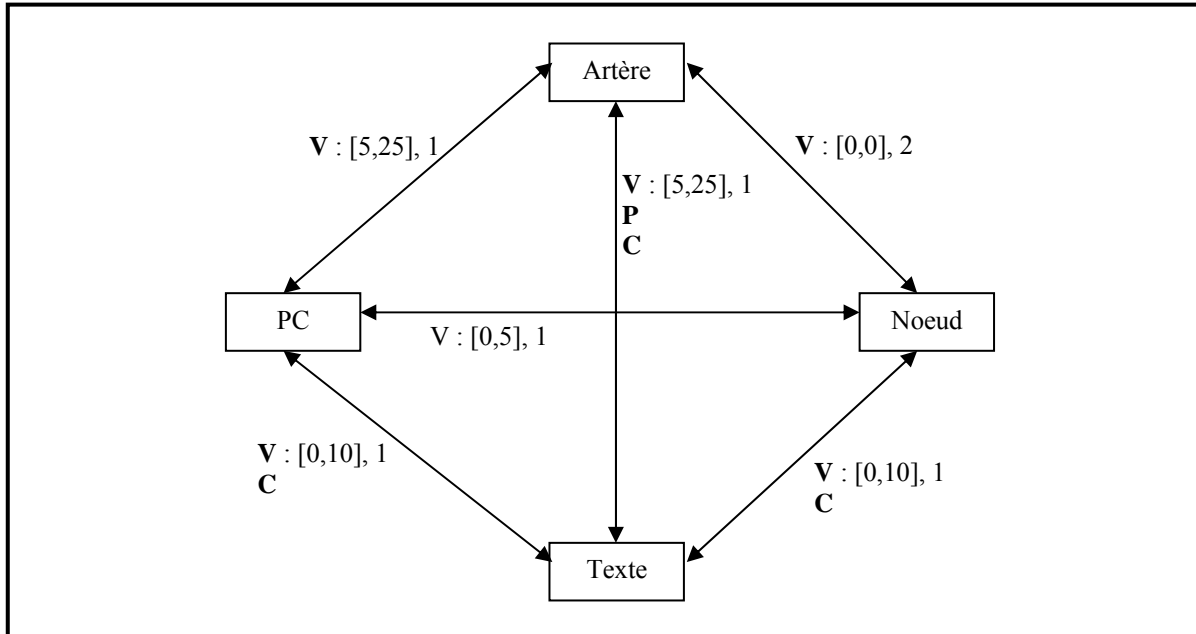
- Le second niveau est le type de représentation des objets. Il est en effet possible de catégoriser les différentes représentations des objets lorsqu'ils sont élémentaires. Nous avons identifié quatre principaux « types » d'objets :
  - Les symboles, que nous séparons des caractères lorsqu'ils peuvent être définis par l'agencement d'entités plus élémentaires (un PC peut être défini à partir de formes géométriques plus ou moins élémentaires, une porte est un arc associé à un vecteur, une note de musique est la boucle associée à un vecteur...). (Type A)
  - Le linéaire, qui est constitué de lignes non fermées, c'est par exemple le cas du réseau sur nos documents, des portées sur les partitions musicales ... (Type B)
  - Les polygones, utilisés par exemple dans le cas des murs sur les plans architecturaux, des contours de pièces pour les schémas mécaniques... (Type C)
  - Le texte (Type D)

Chaque objet élémentaire du domaine défini par l'utilisateur doit appartenir à l'un de ces groupes. Dans le cas des documents réseau, les PC et les nœuds d'infra sont des symboles (**type A**) alors que les artères sont du type linéaire (**type C**). A partir de ces types d'objets pré-définis, il est alors possible d'énumérer un ensemble de relations classiques :

- Le voisinage : les 4 types peuvent être associés par ce lien. Les propriétés de cette relation sont l'intervalle de distance et la cardinalité. Elle est notée **V**, elle est bi-directionnelle.
- La colinéarité ou parallélisme : concerne les types B, C et D. La propriété de cette relation est la tolérance d'angle. Elle est notée **P**, elle est bi-directionnelle.
- La caractérisation : elle concerne le Type D, associé à n'importe quel autre type. La propriété de cette relation est la grammaire associée à l'objet caractérisé par le texte. Cette relation est notée **C**. Elle est dans une seule direction, vers l'objet de type D. Notons qu'il est possible de spécialiser cette relation de caractérisation qui peut avoir différents buts : l'identification, la cotation, la localisation géographique...

- L'inclusion : elle concerne le type C, associé à n'importe quel autre type. (**noté I**)

Sur la Figure 3.23, les différentes relations sont représentées à l'aide d'une unique flèche, pour des questions de lisibilité. En terme d'implémentation, ce n'est pas le cas, chaque type de relation génère un arc différent.



*Figure 3.23 : Les relations existant entre objets élémentaires sur les plans de réseau*

### 3.5.2.2. L'acquisition des connaissances du domaine

Afin d'acquérir ces connaissances du domaine, une première interface a été développée. Elle n'est pas complète mais elle permet de montrer la faisabilité des idées exposées ci-dessus. Nous illustrons cette interface sur la Figure 3.24. L'utilisateur est guidé tout au long du processus d'acquisition par l'intermédiaire de messages lui proposant les différentes alternatives possibles. Dans la partie basse, l'utilisateur peut définir les propriétés de la relation de voisinage entre les PC et les nœuds d'infra. Ces propriétés sont celles énumérées ci-avant.

En sortie de cette interface, un fichier décrivant le graphe est généré. Ce fichier représente de façon statique les connaissances concernant les objets du domaine et les relations unissant ces objets.

Néanmoins, puisque notre objectif est l'interprétation des documents représentant ces objets, une phase d'opérationnalisation est nécessaire. Cette phase a pour but de structurer et de paramétrer le système d'interprétation en utilisant les différents objets et leurs propriétés ainsi que les relations entre eux.

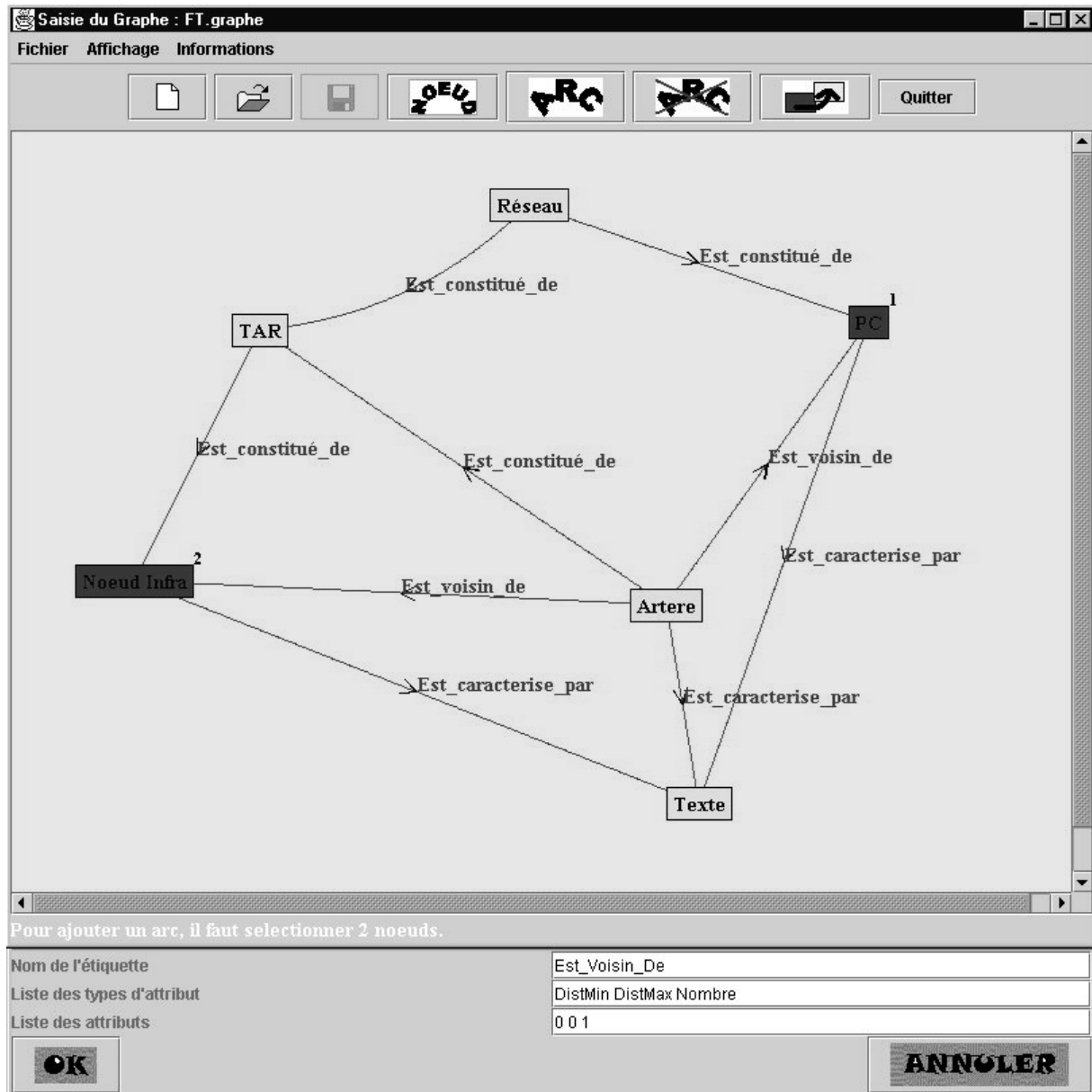


Figure 3.24 : L'interface d'acquisition du réseau pour la représentation des connaissances du domaine

### 3.5.2.3. Vers la génération automatique du SMA

L'objectif de cette partie est de présenter nos contributions concernant la génération automatique de la partie « domaine » du système multi-agents d'interprétation, à partir du modèle acquis par l'interface. A partir de l'image du document, l'objectif de l'interprétation est d'instancier certains objets présents sur le document, en fonction de l'intention de l'utilisateur. Dans ce cadre, les relations de composition dans le réseau sont utilisées pour structurer le système en couches. Les autres relations sont quant à elles exploitées pour la construction des objets lorsque ce sont des constituants, et pour la validation de la cohérence pour les objets de la même couche. Les différentes étapes exposées ci-dessous ne sont pas encore implémentées, mais nous montrons leur faisabilité.



- La structuration du système en couches

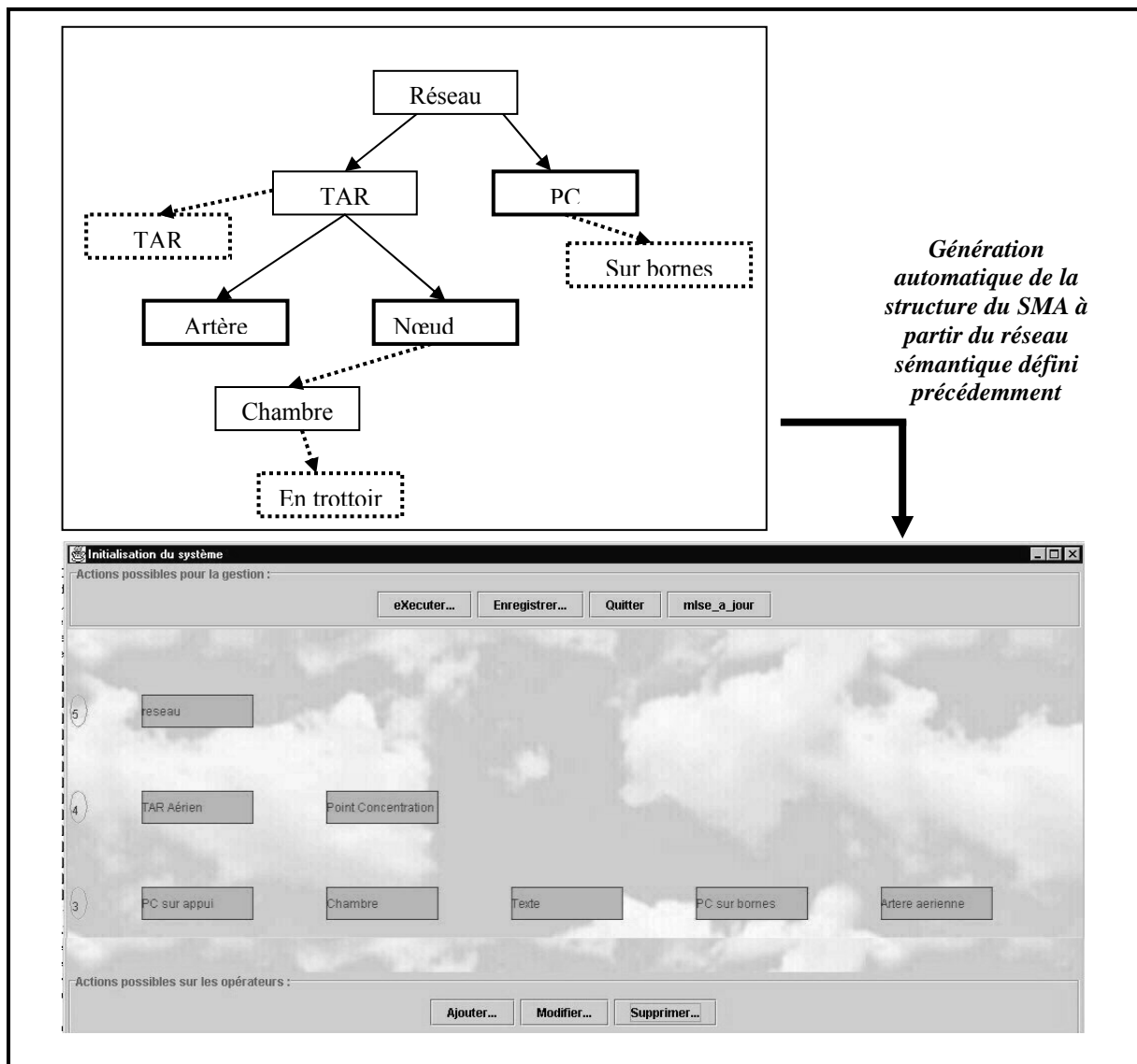


Figure 3.25 : Illustration de la génération automatique de la structure du SMA

Cette première phase a pour but de répartir les agents constructeurs de données sur différentes couches en fonction des liens de composition et de spécialisation. Par analyse des relations présentes dans le réseau, il est possible de déduire une structure hiérarchique. Nous illustrons ce point sur la Figure 3.25.

- La génération de constructeurs pré-définis

Parmi les relations entre types d'objets évoquées en 3.5.2.1, certaines correspondent, en phase d'interprétation, à une reconstruction simple et valable quel que soit le document pris en compte. La relation de voisinage en est une parfaite illustration. Il est possible de concevoir un agent générique qui permette l'association d'un symbole et de la chaîne de caractères associée par relation de voisinage. Il en est de même pour l'inclusion ou la colinéarité. Nous n'avons pas mis en œuvre de tels

opérateurs mais nos travaux actuels concernent la définition de points représentatifs des objets et de relations spatiales génériques. Ces travaux sont proches des travaux menés dans le cadre du projet ICARE à l'IRIT de Toulouse [Matsakis 1998] [Desachy 1998]. L'approche de cette équipe consiste à utiliser des techniques d'estimation des relations spatiales entre objets, en exploitant les histogrammes de forces. Notons que cette technique est aussi utilisée dans [Tabbone 2001] pour l'extraction de signatures de symboles. En définissant de telles relations génériques, le système pourra, en exploitant les liens transversaux contenus explicitement dans le réseau sémantique générer automatiquement le scénario d'interprétation.

Reprenons pour illustrer ces propos l'exemple des points de concentration évoqué à plusieurs reprises précédemment. L'agent *PC* intégré au système suite à la génération automatique de la structure du SMA peut, en consultant le réseau sémantique, savoir qu'il doit disposer des *PC\_graphiques* et de *toponymes* pour construire les objets *PC*. Cela lui permet de gérer la sous-traitance en utilisant le protocole des réseaux contractuels. Puis, à partir d'un constructeur générique d'association symbole / chaîne et des paramètres contenus dans le réseau sémantique, il pourra reconstruire les PC. Le scénario aura alors été automatiquement généré jusqu'à la reconnaissance des éléments élémentaires. Nous reviendrons sur ce point lors des perspectives.

- Les validations de cohérence

Pour la validation de la cohérence des objets, la stratégie consiste à exploiter les relations qui n'ont pas été utilisées pour la construction de la donnée. Dans le cadre de l'application *NATALI v2*, les objets sont reconstruits en utilisant la topologie des éléments constitutifs, le plus souvent le symbole et la chaîne de caractères associée par relation de voisinage. Dans ce cadre, nous exploitons, pour la validation de la cohérence des données, la grammaire des objets de type texte associé. En effet, sur nos documents réseaux, chaque objet est associé à un toponyme respectant une grammaire particulière. Il est donc possible de vérifier l'adéquation de la reconnaissance du symbole et de la reconnaissance du texte. Dans le cas du réseau, les chaînes de caractères associées contiennent généralement la longueur du câble, nous envisageons donc l'exploitation des informations de type cotations. Ce point est important puisque cette relation de cotation est souvent présente sur d'autres types de documents, comme les schémas mécaniques, les plans architecturaux...

Notons également que de ce point de vue validation de cohérence, les éventuelles bases de données existantes constituent des sources d'informations importantes. Dans le cadre de *NATALI* par exemple, nous aurions pu disposer, si le projet avait continué, de la liste des identifiants de Point de Concentration par zones géographiques. Cela aurait pu être utilisé pour valider la reconnaissance de ces PC. C'est également le cas pour la plupart des documents géoréférencés, puisque des bases de données sont de plus en plus souvent disponibles avec les noms de villes, de rues...

### **3.5.3. Les connaissances pour la reconnaissance des objets**

Dans cette section, nous présentons notre contribution concernant la représentation des connaissances pour la reconnaissance des formes dans le système *NATALI v2*. L'objectif de la représentation et de l'opérationnalisation de ces connaissances est de doter le système de capacités adaptatives et évolutives pour reconnaître les objets identifiés comme élémentaires au niveau du domaine. Nos propositions seront illustrées sur le problème de la reconnaissance de caractères et de symboles.

#### **3.5.3.1. Un modèle de la reconnaissance d'une forme**

Dans la première partie du document, nous avons abordé l'un des problèmes d'une chaîne d'interprétation : celui de la reconnaissance de caractères et de symboles. Un modèle de stratégie de reconnaissance, orienté tâche, a alors été proposé (figure 2.3.). Ce modèle est basé sur le fait que la reconnaissance d'une forme exige souvent l'extraction d'une représentation de celle-ci différente de l'image, permettant d'en extraire l'information discriminante (cf. Partie 2). Un tel modèle manipule donc des connaissances que nous avons précédemment qualifiées de stratégiques, mais aussi des connaissances descriptives. Ces deux aspects sont détaillés par la suite.

##### *i. Considérations stratégiques*

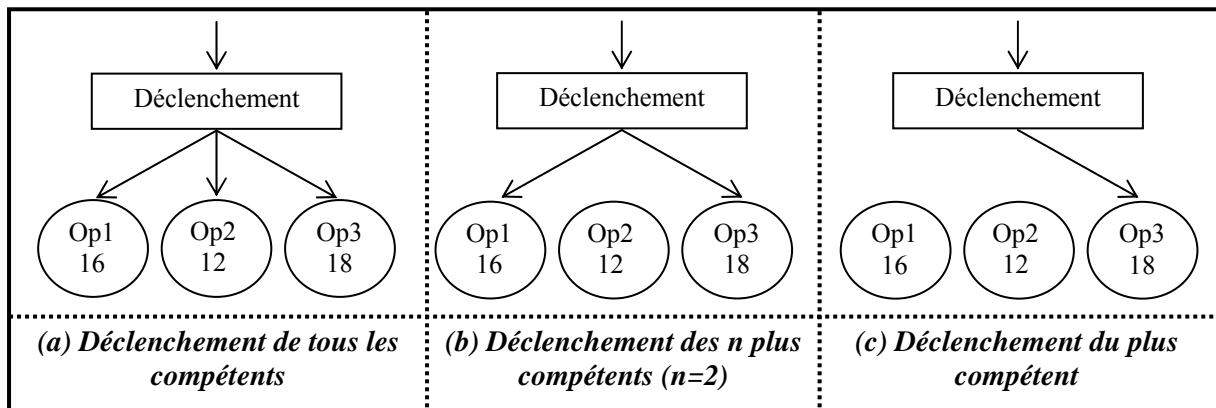
En terme de stratégies, l'arbre de tâche présenté sur la figure 2.3. exploite un unique modèle de représentation de la forme. Or, bien souvent, la fiabilité d'une reconnaissance exige la mise en jeu de points de vue différents. Nous avons donc étendu ce modèle à la possibilité de multiplier les points de vue. Pour pouvoir mettre en œuvre ce principe de la co-exploitation de ressources, nous avons implémenté trois agents ayant pour but la reconnaissance de caractères et de symboles. Les principes utilisés par ces trois agents (extraction d'invariants à partir de la transformée de Fourier Mellin, des moments de Zernike, et des sondes circulaires puis classification) sont décrits dans la première partie du manuscrit. A partir de différents points de vue, deux théories d'exploitation sont souvent distinguées :

- La fusion d'informations, lorsque les points de vue sont des données. La fusion est définie comme « un procédé d'intégration d'informations provenant de sources multiples en vue de produire des données unifiées les plus spécifiques et les plus pertinentes sur une entité, activité ou événement » [Garbay 1998]. La fusion d'informations constitue actuellement un domaine de recherche très actif, à la fois pour fusionner des données symboliques, des données numériques ou encore une association des deux. La conférence FUSION, le groupe de travail très actif du PRC ISIS (GT 6) dédié à cette thématique, et les très nombreuses publications depuis plus de dix ans témoignent de l'activité de cette communauté [Bloch 2000][Cassou-Ribehart 1999][Eallet 2000].

- La compétition, lorsque les points de vue sont les opérateurs. La compétition est définie comme « la mise en œuvre simultanée d’opérateurs susceptibles de produire des informations multiples et éventuellement conflictuelles sur une entité ». Là encore, la communauté de l’interprétation est très active dans ce domaine, mais notons que le but de ces compétitions est souvent limité à la segmentation d’images [Charroux 1996], [Desachy 1999], [Dhéréte 1999].

Ces deux notions sont complémentaires. En effet, dans le cas de la fusion, « *les sources multiples* » sont issues d’opérateurs concurrents, et dans le cas de la compétition, les « *informations multiples* » produites doivent être fusionnées pour prendre une décision finale. Il s’agit donc dans les deux cas de multiplier les points de vue, qu’ils soient descriptifs ou opératoires. Dans notre système, dont la stratégie est basée sur le protocole du réseau contractuel, ces deux niveaux se retrouvent dans :

- Le choix des agents déclenchés. Le protocole du réseau contractuel que nous avons mis en place permet de choisir, lorsque plusieurs agents sont compétents pour répondre à une tâche donnée, d’en déclencher un ( c ), plusieurs ( b ) ou tous ( a ) (Figure 3.25) ;

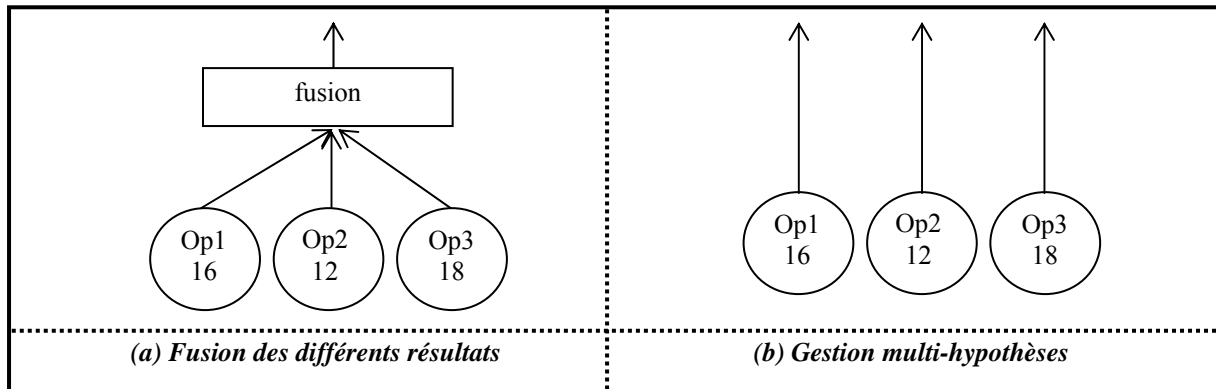


*Figure 3.26 : Les stratégies possibles de déclenchement d’opérateurs*

- L’exploitation des résultats. Lorsque plusieurs agents ont été déclenchés, deux stratégies peuvent être envisagées : la première consiste à transmettre les résultats de chacun à l’agent supérieur et à les conserver afin de prendre une éventuelle décision à un niveau plus élevé de la hiérarchie. C’est une gestion multi-hypothèses des résultats. La seconde solution opte en revanche pour une fusion de ces résultats dès le niveau demandeur (Figure 3.26).

Dans le cadre de notre travail, nous avons essentiellement abordé le premier point, en permettant au système d’adopter les différentes stratégies de la Figure 3.25 et en utilisant la stratégie illustrée par la partie (b) de la Figure 3.27 pour la gestion des résultats. Nous multiplions ainsi les hypothèses possibles afin de prendre la décision à haut niveau, dans les couches correspondant aux connaissances du domaine (3.5.2.1). Nous envisageons néanmoins, dans le cadre de perspectives, l’intégration

d'outils génériques de fusion d'informations puisque ce thème constitue l'un des axes de recherche du laboratoire [Lefèvre 2000].



*Figure 3.27 : Les stratégies possibles de gestion des résultats*

Le choix des agents à déclencher est basé sur le protocole du réseau contractuel. Contrairement à *NATALI v1*, des « scores » de compétences sont associés aux différents agents dans *NATALI v2*, comme le montre la Figure 3.27. Ces scores constituent une partie des connaissances stratégiques du système pour la reconnaissance de formes puisqu'ils permettent l'élaboration dynamique d'enchaînement de traitements. De plus, l'utilisateur peut lui-même décider si chacun des agents doit opter pour la stratégie de « tous les compétents », des « n plus compétents » ou du « plus compétent ». Ce choix peut par exemple être basé sur le compromis fiabilité/rapidité souhaité, il va déterminer la stratégie globale du système. En vue de rendre le système adaptatif et évolutif, il est important que les scores de compétences puissent :

- Etre initialisés en fonction du type de document. Pour cela, une interface permet, entre autres, l'initialisation de ces scores de compétence, elle fera l'objet du paragraphe 3.5.3.2 ;
- Evoluer au cours de la vie du système. Si les exemples ont été mal choisis en phase d'acquisition, il est important que les scores de compétences soient évolutifs. Nous présenterons en 3.5.3.3 un mécanisme possible d'évolution des scores.

Notons que, même si cela n'a pas encore été mis en œuvre, l'une de nos récentes préoccupations consiste à intégrer la dépendance entre le score d'un agent, les pré-traitements utilisés et le contexte de traitement. L'extraction d'un modèle de représentation de forme exige souvent en effet un traitement préalable des images, lui-même dépendant du contexte (binarisation, extraction de contours, squelettisation ...). Ces traitements sont d'une grande influence sur la qualité des résultats et leur efficacité passe la plupart du temps par un réglage de nombreux paramètres. Ce point rejoint la nécessité de modéliser les connaissances opératoires de traitement d'images, comme nous l'avons évoqué à plusieurs reprises. En effet, pour pouvoir régler efficacement les paramètres d'un algorithme, il est fondamental de connaître le contexte dans lequel celui-ci va être utilisé. Une solution pourrait

être d'effectuer des mesures telles que le bruit sur l'image ou la densité de formes et de représenter le lien entre ces mesures et le paramétrage à donner aux traitements [Lecoq 2001]. Dans ce contexte, l'intégration de l'incomplétude et de l'imprécision des informations est indispensable [Lefèvre 1994].

*ii. Considérations descriptives*

Sur le modèle de la figure 2.3 apparaissent également des connaissances que nous avons précédemment qualifiées de descriptives. En effet, lorsqu'il s'agit d'« *appairer l'instance du modèle extrait avec une base de modèles* », il est nécessaire de disposer de cette base de modèles. Elle constitue la connaissance dont dispose le système sur la représentation graphique des formes présentes sur le document analysé. Dans ce contexte, les contraintes sont les mêmes que celles évoquées ci-avant pour les connaissances stratégiques, puisque les connaissances descriptives doivent elles aussi :

- Être initialisées en fonction du type de document. L'interface présentée en 3.5.3.2 apportera des éléments de solution quant à ce problème ;
- Pouvoir évoluer au cours de la vie du système. En effet, généralement, la fiabilité du système de reconnaissance est étroitement corrélée à la richesse de sa base de modèles. Nous présenterons quelques pistes concernant l'évolution de ces bases en 3.5.3.3. Nous n'apportons néanmoins pas de solution au problème de l'apprentissage incrémental du point de vue classification.

**3.5.3.2. L'acquisition des connaissances pour la reconnaissance des formes**

Ci-avant, nous avons montré que les connaissances concernant la reconnaissance de formes étaient dépendantes du type de document traité. Elles doivent donc être acquises lors de l'initialisation du système. Elles sont de deux principaux types :

*i. L'acquisition des connaissances descriptives*

Ce point concerne essentiellement la représentation graphique des concepts du domaine représentés sur les documents. Il s'agit alors, compte tenu du modèle de reconnaissance adopté, de la base de modèles (ou base d'apprentissage) utilisé par le processus de reconnaissance. Nous avons développé une interface permettant cette acquisition. Son objectif est qu'un utilisateur « naïf » du système puisse décrire les objets à reconnaître sur le document du point de vue graphique, après en avoir donné un point de vue « métier » (3.5.2.2). Pour ce faire, deux principaux types d'approches sont possibles :

- Par une description structurelle / syntaxique des formes. Lorsque de telles approches sont adoptées, l'utilisateur doit alors décrire ses objets au moyen de graphes ou plus souvent de grammaires. C'est par exemple le cas dans [Ah Soon 1998], [Dosch 2000] [Garcia 2001] ou encore [Couasnon 2001]. Néanmoins, la définition d'une telle grammaire exige évidemment une

bonne connaissance du système de reconnaissance. A titre d'indication, l'auteur de [Coüasnon 2001] précise lui-même que l'initialisation de son application « reconnaissance de partitions musicales » avait demandé un travail important pour la définition de la grammaire correspondante.

- Par un apprentissage par l'exemple. Dans ce cas, l'utilisateur apprend au système la représentation graphique des objets qu'il faut reconnaître en montrant un(des) exemple(s). Ce genre d'approche n'existe pas à notre connaissance dans la littérature sur l'interprétation de documents techniques, mais des exemples existent pour l'analyse de documents structurés [Brugger 1996].

Nous avons opté, dans le cadre de *NATALI v2* pour ce second type d'approche. Une interface permettant à son utilisateur d'acquérir les connaissances descriptives graphiques a été développée. Cette interface prend en compte le fait qu'un unique exemple ne suffit pas à constituer une base modèle suffisante. Aussi, une fonctionnalité permettant la recherche de formes ressemblantes est-elle intégrée. Après cette recherche, l'utilisateur valide ou non les formes considérées comme ressemblantes. Cela permet de pallier le fait que l'utilisateur ne choisit pas nécessairement la forme la plus représentative pour un apprentissage.

L'acquisition du modèle de représentation d'une forme se déroule en 7 étapes :

- L'utilisateur choisit une image contenant la forme à apprendre (1) ;
- Il saisit une imagerie de sa forme, à l'aide de la souris (2) ;
- Il peut isoler la forme en question en utilisant un outil de gommage (3) ;
- Il lance la recherche de formes ressemblantes, sur la même image ou sur une autre (4) ;
- Les formes ressemblantes (au sens des différents modèles) lui sont alors proposées (5) ;
- L'utilisateur peut supprimer des sur-détections ou ajouter des formes oubliées (6) ;
- L'export des résultats génère la base des modèles (7) ;

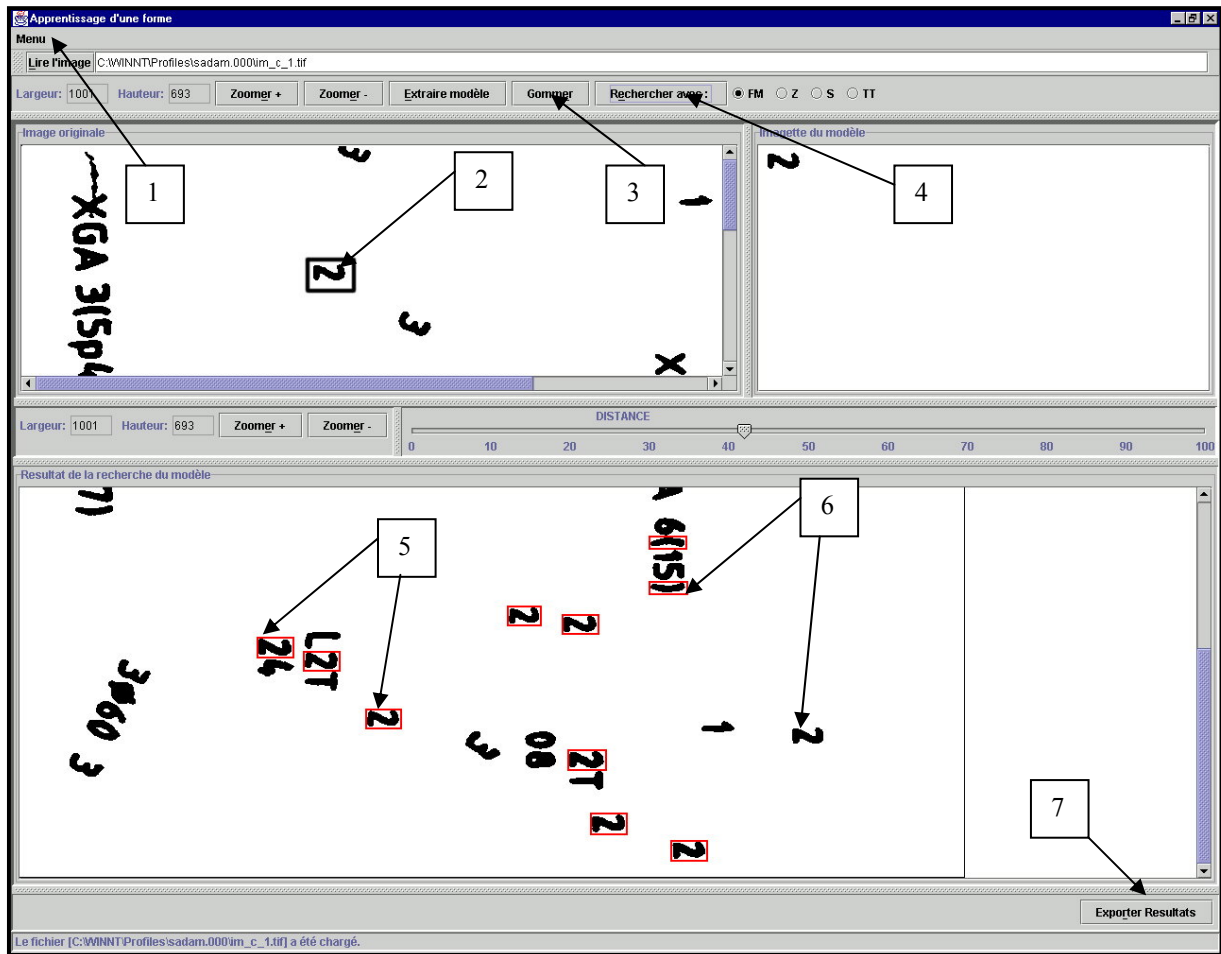


Figure 3.28 : L'interface d'acquisition des connaissances descriptives graphiques appliquées aux caractères

Sur la Figure 3.28, l'interface est présentée pour un problème de reconnaissance de caractères. Cela est simplement dû au fait que nous ne disposons pas d'image contenant suffisamment de symboles pour montrer les différentes fonctionnalités. Notons que l'interface peut être assez facilement utilisable par un naïf puisque l'utilisateur est guidé au travers des différentes étapes et que cette aide n'intègre pas de notion complexe de traitement d'images. Elle peut en outre être utilisée pour l'étiquetage de base de caractères. La recherche de formes ressemblantes permet en effet de gagner un temps considérable puisque l'utilisateur n'a plus qu'à valider ou invalider les résultats au lieu de rechercher sur l'image les différentes occurrences de la forme. Notons que cette approche est parfaitement en adéquation avec les contraintes du projet *DocMining* puisque l'application peut rapidement être initialisée.

## ii. L'acquisition des connaissances stratégiques

L'utilisation du protocole des réseaux contractuels impose que les différents agents soient en mesure d'évaluer leurs capacités à résoudre un problème donné. Cette connaissance concernant ces scores d'aptitude est fortement dépendante du document traité. Elle doit donc faire l'objet d'une procédure d'acquisition lors du passage à une nouvelle application. Pour ce faire, nous utilisons la même interface que celle présentée en (i.). Après la recherche de formes ressemblantes, l'utilisateur apporte



une validation, qui peut être considérée comme la « vérité terrain ». Il est alors possible de calculer des statistiques correspondant à des scores de reconnaissance obtenus par chacun des agents. Pour ce faire, nous considérons le problème comme un problème de classification à deux classes :

- La classe correspondant à la forme cherchée, notée **C** ;
- La classe correspondant à un rejet, notée **R**.

Le classifieur prend une décision binaire, les formes sont donc affectées à l'une ou l'autre de ces deux classes. Nous notons :

- $c$  ou  $r$  : la décision prise par le classifieur ;
- $N_{lues}$  : le nombre de formes affectées à la classe recherchée ;
- $N_{rejetées}$  : le nombre de formes affectées à une classe de rejet ;
- $N_{totales}$  : le nombre total de formes sur l'image, tel que :

$$N_{totales} = N_{lues} + N_{rejetées} \quad (98)$$

Après l'étape de validation/correction, il est possible décomposer  $N_{lues}$  et  $N_{rejetées}$  en :

$$N_{lues} = N_{lues\ validées} + N_{lues\ corrigées} \quad (99)$$

$$N_{rejetées} = N_{rejetées\ validées} + N_{rejetées\ corrigées} \quad (100)$$

Où :

- $N_{lues\ validées}$  est le nombre d'éléments de  $\{c/C\}$
- $N_{lues\ corrigées}$  est le nombre d'éléments de  $(c/R)$
- $N_{rejetées\ validées}$  est le nombre d'éléments de  $(r/R)$ .
- $N_{rejetées\ corrigées}$  est le nombre d'éléments de  $(r/C)$

A partir de ces différentes valeurs, il est possible de mesurer les performances de l'agent à deux niveaux :

### *i. Au niveau global*

Les performances de l'agent sont alors celles d'un classifieur. Elles peuvent s'exprimer par :

- Le taux de reconnaissance de l'agent :

$$T_{reconnaissance} = \frac{N_{lues\ validées} + N_{rejetées\ validées}}{N_{totales}} \quad (101)$$

- Le taux de fausse alarme de l'agent :

$$T_{fausse\ alarme} = \frac{N_{lues\ corrigées}}{N_{totales}} \quad (102)$$

- Le taux de non détection de l'agent :

$$T_{non\,detection} = \frac{N_{rejetées\,corrigées}}{N_{totales}} \quad (103)$$

**ii. Au niveau classe**

Les performances sont alors exprimées pour chacune des deux classes **C** et **R**. avec :

- Le taux de reconnaissance pour la classe recherchée :

$$T_{reconnaissance\,/\,classe\_cherchée} = \frac{N_{lues\,validées}}{N_{lues\,validées} + N_{rejetées\,corrigées}} \quad (104)$$

- Le taux de confusion pour la classe recherchée :

$$T_{confusion\,/\,classe\_cherchée} = \frac{N_{lues\,corrigées}}{N_{lues\,validées} + N_{rejetées\,corrigées}} \quad (105)$$

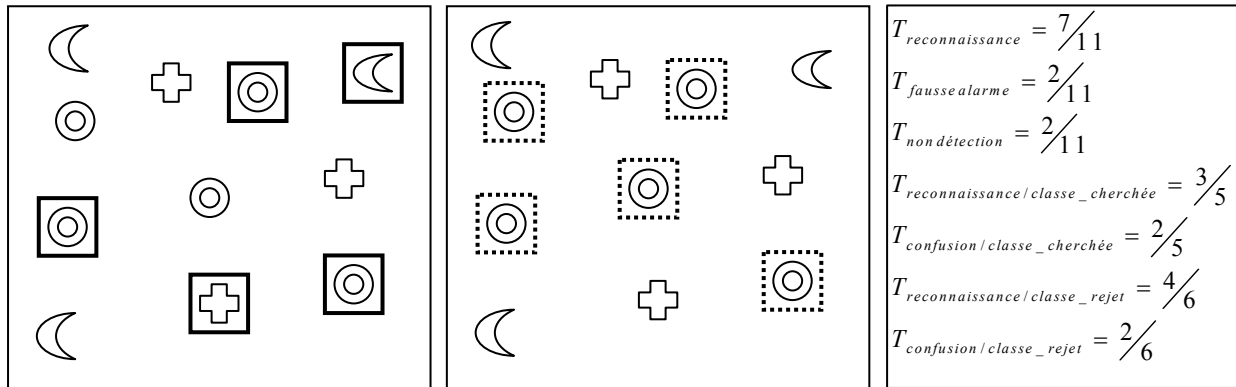
- Le taux de reconnaissance pour la classe rejet :

$$T_{reconnaissance\,/\,classe\_rejet} = \frac{N_{rejetées\,validées}}{N_{rejetées\,validées} + N_{lues\,corrigées}} \quad (106)$$

- Le taux de confusion pour la classe rejet :

$$T_{confusion\,/\,classe\_rejet} = \frac{N_{rejetées\,corrigées}}{N_{rejetées\,validées} + N_{lues\,corrigées}} \quad (107)$$

Les différentes mesures de performances sont illustrées en Figure 3.1, pour un exemple jouet. La forme à apprendre pour cet exemple est l'anneau. Le cadre de gauche représente les résultats issus de la recherche de formes ressemblantes. Les formes associées à la classe cherchée sont encadrées par un carré en gras. Deux formes ont été attribuées à tort à cette classe, alors que deux formes de la classe anneau n'ont pas été détectées. Dans le cadre du milieu sont illustrés les résultats à l'issue de la phase de correction, lors de laquelle a lieu l'évaluation. A droite, nous donnons les scores correspondants.



*Figure 3.29 : Un exemple jouet, avant et après correction et les performances correspondantes*

Dans notre contexte, il est important de minimiser les oublis. En effet, lorsque les données sont transmises au niveau du domaine (Figure 3.27), il est alors possible d'éliminer les sur-détections par intégration des connaissances « métier ». C'est pourquoi nous utilisons comme score d'évaluation le taux de reconnaissance de la classe recherchée,  $T_{\text{reconnaissance/classe recherchée}}$ . Il faut néanmoins préciser que ce choix est lié au contexte de notre système, d'autres mesures telles que le coût d'édition pourraient être utilisées [Chhabra 1999] si le système était limité à l'étape de reconnaissance, sans qu'il y ait d'étapes ultérieures de validation. Le coût pourrait alors être défini comme la somme des deux taux de confusions par classe ((105) et (107)). De plus, la prise en compte de classes préalablement apprises pourrait également minimiser les corrections à apporter.

Ces statistiques sont utilisées comme initialisation des scores de compétences des agents ayant pour objectif la reconnaissance du texte et des symboles. Ces scores sont ainsi adaptés aux données, mais aussi au type de document les contenant. Avec ces travaux, nous rejoignons une double problématique très actuelle dans la communauté de l'interprétation : celle de l'évaluation de performances et de l'adaptation d'opérateurs. Nous avons déjà évoqué ce problème à l'occasion de la discussion dans la première partie mais notons qu'ici, notre contribution apporte une solution originale dans le cadre des opérateurs de reconnaissance, en adaptant le score des opérateurs aux données, et au document traité. Des travaux sont actuellement en cours dans notre équipe pour adopter la même démarche pour la caractérisation d'opérateurs de vectorisation, en proposant des résultats obtenus par différents algorithmes à un utilisateur qui, par ses éventuelles corrections apporte la « vérité terrain ». Cela nous permet de caractériser les performances de l'algorithme, en tenant compte de l'intention et de la précision souhaitée par l'utilisateur.

### **3.5.3.3. La mise en œuvre dans le SMA**

Outre le processus d'acquisition mis en place, il a également été nécessaire d'apporter des modifications au sein même du moteur du système pour intégrer les connaissances évoquées ci-avant dans le système *NATALI v2*. De ce point de vue, nos contributions se situent à trois principaux niveaux :

- L'intégration dans le système de deux nouveaux agents, capables de reconnaître des caractères et des symboles. Cette tâche fut facilitée par la ré-utilisation de certains modules génériques, comme celui de communication qui avait été développé dans le cadre de *NATALI v1*.
- L'adaptation du système pour qu'il puisse mettre en œuvre concrètement le protocole du réseau contractuel. Cela a consisté à re-développer le module permettant la sous-traitance pour que celui-ci prenne en compte les trois stratégies possibles (le plus compétent, tous les compétents, les n plus

compétents). Le module chargé de l'interprétation des messages a, lui aussi, évolué pour que les messages entre agents puissent véhiculer le score d'aptitude.

- Une autre modification importante apportée au système *NATALI v1* a pour but l'évolution des deux principaux types de connaissances évoqués ci-avant. En effet, si l'étape d'acquisition permet une première initialisation des connaissances, celle-ci peut bien souvent se trouver incomplète, voire erronée. Il est donc nécessaire que ces connaissances évoluent au cours de la vie du système. Pour ce faire, deux axes sont envisageables : avoir recours à une phase de validation des résultats par l'utilisateur ou attendre que des données cohérentes aient été reconstruites plus haut, lorsque les connaissances du domaine sont utilisées (typiquement, le fait qu'un symbole donné ait été associé à une chaîne de caractères cohérente au regard de ce symbole). Quel que soit le cas, il est nécessaire de connaître la chaîne de reconnaissance qui a été utilisée afin de faire évoluer le score des agents ayant été déclenchés, ainsi que les bases de connaissances descriptives relatives à ces agents. Sur ce point particulier nous avons modifié le système *NATALI v1* afin d'associer aux données générées par le système (qui sont stockées dans la base de données) le scénario qui les a produites. Les données générées sont en quelque sorte encapsulées par les traitements. Néanmoins, en raison des évolutions des besoins en terme d'application (passage de *NATALI* à *DocMining*), nous n'avons pas implémenté de reconstruction complète des objets du domaine. La partie stockage des *scénarii* ayant permis d'obtenir les données est toutefois opérationnelle, et intégrée dans le système.

### **3.6. Conclusion**

Dans cette seconde partie de la thèse, nous avons abordé l'aspect ingénierie des connaissances d'un système d'interprétation de documents techniques. Nous considérons en effet que l'obtention d'un système adaptable et évolutif n'est possible que si les connaissances intégrées au système sont représentées explicitement. Aussi avons nous orienté nos travaux sur la modélisation et la représentation des connaissances. Notre démarche s'est inspirée des travaux menés en interprétation d'images, pour lesquels cette problématique a davantage été prise en compte, et de quelques travaux de référence en ingénierie des connaissances. Dans ce contexte, nous avons cherché à suivre les préconisations de Newell [Newell 1982], en nous plaçant successivement au niveau connaissance, puis au niveau symbole. Au niveau connaissance, nous avons identifié dans le second chapitre, trois catégories de connaissances utilisées par un système d'interprétation : les connaissances descriptives, qui concernent les données manipulées, les connaissances opératoires, qui concernent les traitements utilisés, et les connaissances stratégiques, qui concernent l'ordonnement des traitements. Dans le troisième chapitre, nous avons examiné les choix adoptés par la littérature pour la représentation des connaissances identifiées précédemment, au niveau symbole. Cette étude bibliographique a montré que les différents formalismes connus en intelligence artificielle pouvaient être exploités. Puis, nous avons présenté une analyse d'un premier système développé par notre équipe, du point de vue de l'ingénierie des connaissances. Cette analyse a mis en avant un certain nombre de points forts, comme la généralité des agents développés mais a aussi révélé quelques lacunes. Nous avons donc proposé, dans le cinquième chapitre, quelques pistes de solutions pour aller vers plus de généralité. Ces solutions ont été présentées suivant deux axes :

- La représentation des connaissances du domaine. Dans le premier système, l'importance d'une représentation explicite avait été mise en avant. Néanmoins, aucun formalisme n'avait été intégré et les connaissances étaient représentées de façon procédurale, dans le code des agents. Un utilisateur devait donc, pour définir un nouveau document avec *NATALI v1*, directement intégrer de nouveaux agents dans le système, ce qui requiert de fortes compétences en génie logiciel. Nous avons donc proposé un formalisme explicite de représentation des connaissances du domaine, inspiré des réseaux sémantiques. Une telle explicitation des connaissances a montré différents avantages. D'une part, elle permet une séparation claire entre les connaissances et la façon dont elles sont utilisées pour la reconstruction. D'autre part, elle permet une génération automatique de la stratégie d'extraction des objets du domaine par utilisation de constructeurs génériques, basés sur les relations possibles entre objets. Enfin, elle permet d'intégrer dans le système des contraintes de cohérence sur les objets à extraire. Notons qu'une interface permettant l'acquisition de ces connaissances a été proposée.

- Au niveau des connaissances dites graphiques, nous avons proposé un modèle de stratégie de reconnaissance d'une forme. Elle permet l'utilisation de plusieurs points de vue sur la forme, et donc la génération d'hypothèses multiples sur la présence d'objets dans une zone de l'image. Ce modèle a été opérationnalisé par un système multi-agents, les connaissances descriptives (les bases de modèles) et stratégiques (les scores d'aptitude des différents modèles) étant distribuées au sein de chacun des agents. Là encore, l'accent a été mis à la fois sur l'acquisition des connaissances puisqu'une interface permet l'initialisation des objets et sur leur évolution au cours de la vie du système au travers de messages véhiculant les qualités du scénario qui a été utilisé.

Nous reviendrons sur les nombreuses perspectives qu'ouvrent les travaux de cette partie dans la conclusion générale qui suit.



---

# Conclusion Générale et Perspectives

## 4. Conclusion générale et perspectives

Les travaux présentés dans ce mémoire abordent la problématique de l'interprétation de documents techniques. Du fait de la complexité d'un tel problème, ils se trouvent logiquement à la confluence de différentes thématiques de recherche telles que le traitement du signal et des images, la reconnaissance de formes, l'intelligence artificielle, la communication Homme/Machine et l'ingénierie des connaissances. En effet, si ces domaines scientifiques diffèrent dans leurs fondements, ils sont complémentaires et leurs apports respectifs sont indispensables pour la conception d'un système d'interprétation fiable et adaptable. Dans ce contexte pluridisciplinaire, le mémoire est organisé en deux parties.

La première partie regroupe des travaux touchant au domaine de la reconnaissance de formes en exploitant des techniques issues du traitement du signal. Dans ce cadre, notre contribution a consisté à proposer une méthodologie permettant la détection et la reconnaissance de formes multi-orientées et multi-échelles. L'approche adoptée est basée sur la transformée de Fourier Mellin. Cette transformée convergente, complète et invariante permet la reconnaissance de formes isolées, mais aussi, dans une certaine mesure, de formes connectées. Elle offre en outre la possibilité d'estimer l'orientation des formes, ce qui peut s'avérer intéressant lorsqu'il s'agit de caractères qu'il faut grouper en toponymes. Les outils développés ont été évalués sur des bases de synthèse, sur des bases standardisées et sur des bases importantes issues de documents techniques. Les résultats de l'application de cette technique sont très encourageants. Le taux de bonne classification atteint en effet des scores tout à fait compétitifs au regard des approches de l'état de l'art. De plus, ces travaux ouvrent un certain nombre de perspectives qui méritent d'être évoquées dans le cadre de cette conclusion générale.



Certaines d'entre elles sont liées au problème de la classification. Elles ont déjà été largement évoquées dans le cadre d'une discussion (2.4.5). Nous suggérons donc au lecteur de se reporter à cette section pour plus de développements.

Une autre perspective, qui a déjà donné lieu à de premiers résultats, concerne le développement d'un nouvel extracteur de primitives basé sur la transformée en ondelettes. Ce formalisme semble en effet en mesure de proposer des solutions intéressantes au problème de la reconnaissance de formes multi-orientées et multi-échelles [Chen 1999][Shen 1999]. Les premiers résultats obtenus sont encourageants mais nécessitent une formalisation plus complète, ainsi que le développement d'un classifieur adapté à ce type de primitives. Ce classifieur devra permettre la sélection des descripteurs les plus discriminants en fonction des classes.

Un autre modèle de représentation de forme complémentaire est en cours de développement au sein du laboratoire. Il est essentiellement dédié à la reconnaissance de symboles composites par une approche structurale. L'approche, ainsi que de premiers résultats qualitatifs et quantitatifs prometteurs, sont proposés en Annexe B et dans [Adam 2000a][Delalandre 2001]. Les perspectives concernant cet outil y sont également évoquées.

Enfin, concernant cette partie de notre travail, la perspective qui nous semble la plus importante concerne la reconnaissance des formes connectées. Nous avons proposé une approche originale permettant d'appréhender ce problème dans ce mémoire. La méthode s'avère très efficace en terme de taux de reconnaissance mais sa complexité algorithmique lui interdit d'être utilisée sur de grandes portions d'images. Pour pouvoir l'exploiter plus largement, deux axes doivent être envisagés. Le premier concerne la détermination de zones d'intérêt sur lesquelles la méthode peut raisonnablement être appliquée. Nous étudions actuellement la possibilité d'exploiter les résultats de la vectorisation de la partie graphique pour déterminer ces zones. Il s'agit de définir des signatures [Tabbone 2001], rapides à calculer, qui soient caractéristiques de la présence d'un « élément perturbateur ». Le mode « filtrage » de la TFM pourra ensuite être utilisé sur cette zone d'intérêt pour la validation de l'hypothèse générée. Le second axe à développer pour la reconnaissance des formes connectées concerne la parallélisation des traitements implémentant le mode filtrage, afin de diminuer les temps de calcul [Amiot 2000]. L'extraction des invariants de Fourier Mellin correspond à une opération de convolution de l'image avec un banc de filtres. Une première technique a permis de diminuer la complexité algorithmique, il s'agit de l'utilisation de la transformée de Fourier rapide qui permet de transformer le produit de convolution dans l'espace image en un produit dans l'espace des fréquences. Nos travaux actuels portent maintenant sur la possibilité de paralléliser ces calculs.

En guise de bilan sur cette première partie du mémoire, nous pensons qu'au même titre que la plupart des outils bas niveau de l'état de l'art, les traitements présentés peuvent être considérés comme matures, même si les résultats obtenus ne sont pas parfaits. Malgré cette maturité, force est de constater qu'un enchaînement séquentiel et figé de ces briques conduit à des systèmes dédiés à des types de documents particuliers. Cette absence de généralité n'est selon nous pas due aux défauts intrinsèques des outils, mais plutôt à un manque d'explicitation des connaissances intégrées dans les systèmes.

C'est la raison pour laquelle, dans la seconde partie de ce mémoire, nous avons abordé la problématique de l'interprétation de documents techniques avec un point de vue orienté « ingénierie des connaissances ». Les réflexions proposées dans ce cadre, qui sont résumées en (§ 3.6), ont selon nous permis de montrer la faisabilité et la pertinence d'une démarche orientée connaissances pour la conception d'un système d'interprétation. Elles ont donné lieu à une implémentation conduisant à l'évolution de notre premier système *NATALI v1* dans le sens de la généralité. Dans *NATALI v2*, la plus grande partie des connaissances utilisées est en effet représentée de façon explicite. L'architecture logicielle basée sur ces connaissances offre ainsi une bonne adaptabilité à l'évolution des modèles et une grande souplesse au système. Il est possible par l'intermédiaire d'interactions homme-machine, de définir de nouveaux modèles de documents, d'apprendre la représentation graphique des objets qui y sont représentés et de générer dynamiquement les stratégies d'interprétation qui découlent de cette définition.

Toutefois, force est de constater que ces premiers travaux orientés ingénierie des connaissances n'ont pas résolu le problème extrêmement complexe de la conception d'un système d'interprétation complètement général. De très nombreuses questions restent ouvertes et constituent selon nous les perspectives à aborder.

La première concerne un point que nous avons évoqué à plusieurs reprises dans le manuscrit : la représentation des connaissances opératoires. Nous avons montré que *NATALI V2* était en mesure de choisir le bon modèle de représentation des formes, grâce au module d'acquisition des connaissances graphiques. Toutefois ce module n'effectue aucune transformation préalable des images et n'utilise que des images binaires en entrée. Or, dans bien des cas, l'obtention d'une image binaire « propre » passe par un ensemble de traitements préalables (segmentation, lissage, morphologie mathématique...). Les résultats de ces traitements sont soumis au réglage de paramètres dont l'influence peut être considérable. Ce vaste problème du réglage des paramètres constitue actuellement l'une des priorités de la communauté scientifique du domaine. Il vise à expliciter l'expertise des spécialistes de traitement d'image qui sont capables, au moins partiellement, d'effectuer cette tâche.

Nous pensons qu'il est nécessaire de répondre à trois questions fondamentales pour proposer une automatisation de celle-ci :

- **Comment caractériser un contexte de traitement ?** Deux points nous semblent fondamentaux dans ce cadre : l'exploitation d'une analyse multi-résolution [Ogier 2000a] qui permet bien souvent d'extraire des indices globaux sur l'image, et la prise en compte de formalismes de représentation imprécis et incertains [Lefevre 2000]. Ce second point doit d'ailleurs être envisagé dans l'ensemble de la chaîne, tant les données manipulées par un système d'interprétation d'images sont peu fiables.
- **Quels sont les paramètres qui influent sur les performances d'un algorithme ?** Il s'agit ici d'abord d'un travail de spécification des algorithmes de traitement. Ce travail, qui peut paraître relever du détail, est en réalité extrêmement important. A l'heure de « *l'open source* », il va dans le sens d'une standardisation et d'un partage de connaissance entre différentes équipes travaillant à travers le monde [Lamiroy 2001]. Il vise à identifier les entrées/sorties des algorithmes, parmi lesquelles il faut ensuite déterminer les paramètres influant sur les résultats des traitements.
- **Comment caractériser les performances d'un opérateur ?** Cette problématique globale de l'évaluation de performances est sans conteste l'une des préoccupations majeures de la communauté depuis près d'une dizaine d'années [Chhabra 1998, 1999][Smeulders 1998]. Ces études sont indispensables pour la caractérisation comportementale des opérateurs en fonction de différents facteurs tels que la sensibilité au bruit par exemple. Toutefois, nous pensons que les évaluations actuelles souffrent d'un défaut non négligeable : elles sont trop décorréées des traitements en amont et en aval et de l'objectif final de leur utilisation. Elles effectuent une analyse (indispensable) à un niveau « microscopique », mais le niveau « macroscopique » consistant à étudier l'influence des interactions entre traitements est mis de côté. Au même titre que les opérateurs particuliers, nous pensons qu'il faut aussi caractériser leurs enchaînements pour répondre à un but donné. Pour illustrer ces propos, prenons l'exemple des performances des algorithmes de binarisation, qui sont totalement corrélées à l'utilisation qui est faite de l'image binaire.

Un second axe de recherche qu'il nous semble important de poursuivre est celui de la communication homme-machine. Dans *NATALI V2*, trois interfaces sont utilisées. Chacune d'elles a fait l'objet d'efforts concernant son ergonomie. Cependant, de nombreux travaux restent à accomplir dans ce domaine pour que celles-ci soient utilisables par un « naïf ». C'est par exemple le cas de l'interface d'expression de l'intention d'interprétation qui requiert une connaissance de la grammaire utilisée pour les communications entre agents. De ce point de vue, nous envisageons d'offrir à l'utilisateur la possibilité d'exprimer ses intentions par l'intermédiaire d'un langage graphique exploitant le réseau

sémantique et l'image elle-même. L'interface d'acquisition des connaissances du domaine demande aussi un certain nombre d'améliorations, essentiellement pour l'expression des relations entre les objets. Une approche par l'exemple, bien que plus difficile à mettre en œuvre que dans le cas des connaissances graphiques, semble une piste à explorer [Dalle 1998]. Une étude préalable de l'expression de ces relations [Dessachy 1998] est donc nécessaire. Enfin, l'interface d'acquisition des connaissances graphiques n'envisage, selon nous, pas suffisamment le fait que l'utilisateur puisse choisir un mauvais exemple pour l'apprentissage des modèles de formes.

Concernant l'opérationnalisation du système, quelques axes de poursuite de ces travaux peuvent aussi être proposés. Citons d'abord la mise en œuvre d'opérateurs génériques de fusion [Dhéréte 1999]. A l'heure actuelle, *NATALI v2* n'est pas en mesure de fusionner les résultats de plusieurs opérateurs coopérants. Tous les résultats cohérents sont gardés dans un esprit « multi-hypothèses ». La mise en œuvre d'opérateurs génériques de fusion d'hypothèses, tant symboliques que numériques, pourrait apporter une robustesse supplémentaire au système. Des opérateurs génériques d'optimisation doivent également être envisagés afin de pouvoir, par exemple, utiliser nos opérateurs en mode « boucle » tant qu'un critère de réussite n'est pas satisfait. Ces opérateurs seront, au même titre que les opérateurs de fusion cités précédemment, transversaux au système en couche. Ils proposeront des services aux différents agents du système, mais sans prendre une place précise dans la hiérarchie.

L'ensemble de ces perspectives définit selon nous les axes des principaux travaux à mener dans les années à venir. Ces objectifs dépassent largement le domaine de l'interprétation de documents techniques, tant du point de vue application (avec une prise en compte de la problématique de l'indexation par exemple), que du point de vue recherche fondamentale. Nous pensons que la satisfaction de ces objectifs passe par la mise en place de collaborations entre équipes travaillant dans différents domaines. L'une des priorités dans ce cadre est selon nous la prise en compte de « l'homme dans la boucle », avec des aspects ergonomiques, linguistiques mais aussi sociaux et culturels pour l'appropriation des systèmes par leurs utilisateurs.



---

# Bibliographie

## Bibliographie

Parmi les nombreuses références bibliographiques citées tout au long du manuscrit, certaines ont constitué des documents de base pour notre travail de thèse. Afin de permettre au lecteur de facilement détecter ces références, nous avons choisi de les mettre en évidence en les signalant par un trait plein sur la gauche. Ceci vise à aiguiller un nouveau chercheur dans le domaine vers ce que nous considérons comme des articles de référence, dans lesquels il pourra trouver les différentes problématiques du domaine et quelques réponses. En outre, un certain nombre de ces références bibliographiques émanent de notre laboratoire. Pour faciliter le repérage de ces nous les avons marqués d'un trait pointillés.

- [Abu Mostafa 1984] Y. S. Abu Mostafa, D. Psaltis, *Recognitive aspects of moment invariants*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 6, pp. 698-706, 1984.
- [Adam 1998] S. Adam, *Contribution à l'interprétation de plans de réseau France Telecom*, **Rapport de DEA** de l'université de Rouen, 1998.
- [Adam 1999a] S. Adam, *Vers une bibliothèque générique de programmation des algorithmes génétiques*, **Rapport Interne PSI**, Disponible à <http://psiserver.insa-rouen.fr/psi/fr/accueil.html>, 1999.
- [Adam 1999b] S. Adam, *Vers une bibliothèque générique de gestion de graphes*, **Rapport Interne PSI**, Disponible à <http://psiserver.insa-rouen.fr/psi/fr/accueil.html>, 1999.
- [Adam 2000a] S. Adam, R. Mullet, J. M. Ogier, C. Cariou, J. Gardes, Y. Lecourtier, *Processing of the connected shapes in raster-to-vector conversion process*, dans **Graphics Recognition – Recent Advances**, édité par A. K. Chhabra et D. Dori, Lecture Notes in Computer Science, Springer Verlag, Vol. 1941, pp. 237-244, 2000.
- [Adam 2000b] S. Adam, J.M. Ogier, C. Cariou, R. Mullet, J. Gardes, Y. Lecourtier, *Combination of invariant pattern recognition primitives on technical documents*, dans **Graphics Recognition – Recent Advances**, édité par A. K. Chhabra et D. Dori, Lecture Notes in Computer Science, Springer Verlag, Vol. 1941, pp. 29-36, 2000.
- [Adam 2001a] S. Adam, J. M. Ogier, C. Cariou, R. Mullet, J. Gardes, Y. Lecourtier, *Utilisation de la transformée de Fourier Mellin pour la reconnaissance de forme multi-orientée et multi-échelle : application à l'analyse automatique de documents techniques*, **Traitement du Signal**, Vol. 18, N°1, pp. 17-33, 2001.
- [Adam 2001b] S. Adam, J.M. Ogier, C. Cariou, J. Gardes, *A Scale and rotation parameters estimator : application to technical document interpretation*, dans les actes de **IAPR International Workshop on Graphic Recognition (GREC)**, Kingston, Canada, pp. 377-380, 2001.

- [Ah Soon 1998] C. Ah Soon, *Analyse de plans architecturaux*, **Thèse de Doctorat** de l'Institut National Polytechnique de Lorraine, 1998.
- [Amiot 2000] F. Amiot, *Vers une architecture parallèle reconfigurable dédiée au traitement d'images et à la vision*, **Thèse de Doctorat** de l'Université de Rouen, 2000.
- [Andrews 1971] H. C. Andrews, *Multidimensional rotations in features selection*, **IEEE Transactions on Computers**, Vol. 20, pp. 1045-1051, 1971.
- [Badreldin 1980] A. Badreldin, A.K.C. Wong, T. Prasad, M. Ismail, *Shape descriptors for N-dimensional curves and trajectories*, **IEEE proceedings on Cybernetics and Society**, pp. 713-717, 1980.
- [Bailey 1996] R. R. Bailey, M. Srinath, *Orthogonal moment features for use with parametric and non parametric classifiers*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 18, pp. 389-396, 1998.
- [Balslev 2000] I. Balslev, K. Doring, R. D. Eriksen, *Weighted central moments in pattern recognition*, **Pattern Recognition Letters (PRL)**, Vol. 21, pp. 381-385, 2000.
- [Bamieh 1986] B. Bamieh, R. J. P. De Figueiredo, *A general moment-invariant/attributed graph method for three dimensional object recognition for a single image*, **IEEE Journal of Robotics Automation**, Vol. 2, pp. 31-41, 1986.
- [Barbey 1999] M. Barbey, *Contribution au développement d'un système d'interprétation de plans de réseau électrique à l'échelle 1/200<sup>ème</sup>*, **rapport de DEA** de l'université de Rouen, 1999.
- [Baujard 1992] O. Baujard, *Conception d'un environnement de développement pour la résolution de problèmes : apports de l'intelligence artificielle distribuée et application à la vision*, **Thèse de Doctorat** de L'université de Grenoble 1, 1992.
- [Bebis 1992] G. N. Bebis, G.M. Papadourakis, *Object recognition using invariant object boundary representations and neural networks models*, **Pattern Recognition (PR)**, Vol. 25, N°1, pp. 25-44, 1992.
- [Belkasim 1988] S. O. Belkasim, M. Shridar, M. Ahmadi, *Optimisation techniques for edge and contour detection*, dans les actes de **SPIE – the international Society of Optical Engineering**, N°956, Dearborn, MI, pp. 9-17, 1988.
- [Belkasim 1991] S. O. Belkasim, M. Shridar, M. Ahmadi, *Pattern recognition with moment invariants : a comparative study and new results*, **Pattern Recognition (PR)**, Vol. 24, pp. 1117-1138, 1991.
- [Bessetes 1998] Vincent Bessetes, Jacky Desachy, *Extraction and classification of urban areas on SPOT images*, dans les actes de **International Geoscience and Remote Sensing Symposium (IGARSS)**, Seattle, USA, pp. 188-196, 1998.
- [Bloch 2000] I. Bloch, *Fusion of numerical and structural image transformation in medical imaging in the framework of fuzzy sets*, dans **Series studies in fuzzyness and soft computing**, Fuzzy systems in medecine, pp. 429-447, 2000.
- [Bokser 1992] M. Bokser, *Omnicolore technologies*, **Proceedings of The IEEE**, Vol. 80, pp. 1066-1078, 1992.
- [Brooks 1981] R. A. Brooks, *Symbolic reasoning among 3-D models and 2-D images*, **Artificial Intelligence Journal**, Vol. 17, pp. 285-348, 1981.
- [Brooks 1983] R. A. Brooks, *Model based three dimensional interpretation of two dimensional images*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 5, N° 2, pp. 140-150, 1983.
- [Brugger 1998] R. Brugger, F. Bapst, R. Inglood, *A DTD extension for document structure recognition*, dans les actes de **Electronic Publishing (EP)**, Saint Malo, France, pp. 343-354, 1998.
- [Cao 1994] J. Cao, M. Ahmadi, S. Shridar, *Handwritten numerals recognition with multiple features and multistage classifiers*, dans les actes de **IEEE International Symposium on Circuits systems**, London, Angleterre, N°6, pp. 323-326, 1994.
- [Casey 1996] R. G. Casey, E. Lecolinet, *A survey of methods and strategies in character segmentation*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 18, N°7, pp. 690-706, 1996.
- [Cassou 1999] D. Cassou-Ribehart, I Bloch, M. Roux, H. Maître, *Résolution contextuelle de conflits entre sources d'informations*, dans les actes du colloque **GRETSI**, Vol. 2, pp. 467-470, Vannes, France, 1999.
- [Charroux 1996] B. Charroux, S. Philipp, J. P. Cocquerez, *Système de vision mettant en œuvre une coopération d'opérateurs de segmentation guidés par l'interprétation*, dans les actes du congrès **Reconnaissance de Formes et Intelligence Artificielle (RFIA)**, Rennes, France, pp. 527-536,

- 1996.
- [Chen 1999] G. Chen, T. D. Bui, *Invariant Fourier-wavelet descriptors for pattern recognition*, **Pattern Recognition (PR)**, Vol. 32, pp. 1083-1088, 1999.
- [Chhabra 1998] A. K. Chhabra, *Graphic symbol recognition : an overview*, dans **Graphics recognition – Algorithms and systems**, lecture notes in computer science, Springer Verlag, Edité par K. Tombre et A. Chhabra, 1998.
- [Chhabra 1999] A. K. Chhabra, I. T. Phillips, *Edit cost as measure of performance of graphic recognition systems*, dans les actes de **International Workshop on Graphic Recognition (GREC)**, Jaipur, India, pp. 331-334, 1999.
- [Clavier 2000] E. Clavier, *Etude de stratégies de tri : application à un système de tri de formulaires*, **Thèse de Doctorat** de l'université de Caen, 2000.
- [Clouard 1995] R. Clouard, C. Porquet, A. Elmoataz, M. Revenu, *Why building knowledge based image segmentation is so difficult*, dans les actes de **International Workshop Knowledge Based Systems for the Reuse of Program Libraries**, Sophia Antipolis, France, pp. 161-185, 1995.
- [Clouard 1999] R. Clouard, A. Elmoataz, C. Porquet, M. Revenu, *BORG : A knowledge based system for automatic generation of image processing programs*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 21, N° 2, pp. 128-144, 1999.
- [Corby 1992] O. Corby, *NMS : Un module multi spécialiste pour le générateur de systèmes expert smeci*, **Rapport Technique**, INRIA, 1992.
- [Cotiasnon 2001] B. Cotiasnon, *DMOS, a generic document recognition method, application to an automatic generator of musical Scores, mathematical formulae and Table Structure Recognition Systems*, dans les actes de **International Conference on Document Analysis and Recognition (ICDAR)**, pp. 215- 220, Seattle, USA, 2001.
- [Crevier 1997] D. Crevier, R. Lepage, *Knowledge-based image understanding systems: a survey*, **Computer Vision and Image Understanding (CVIU)**, Vol. 67, N°2, pp. 161-185, 1997.
- [Dai 1992] M. Dai, P. Baylou and M. Najim, *An efficient algorithm for computation of shape moments from run-length codes or chain codes*, **Pattern Recognition (PR)**, Vol. 25, pp. 1119-1128, 1992.
- [Dalle 1998] P. Dalle, P. Dejean, *Planification en Traitement d'Images : Approche basée sur les données*, dans les actes de **Reconnaissance des Formes et Intelligence Artificielle (RFIA)**, Clermont-Ferrand, pp 75-84, 1998.
- [Daubechies 1990] I. Daubechies, *The Wavelet Transform, time-frequency localization and signal analysis*, **IEEE Transactions on Information Theory**, Vol. 36, pp. 961-1005, 1990.
- [Daubechies 1992] I. Daubechies, *Ten lecture on Wavelets*, CMBS-NSF, **Regional Conference Series in Applied Mathematics**, 1992.
- [Delalandre 2001] M. Delalandre, *Contribution au problème Scan-to-XML*, **rapport de DEA** de l'université de Rouen, 2001.
- [Den Hartog 1996] J. E. Den Hartog, T. K. Ten Kate, J. J. Gerbrands, *Knowledge-based interpretation of utility maps*, **Computer Vision and Image Understanding (CVIU)**, Vol. 63, N°1, pp. 105-117, 1996.
- [Derrode 1999] S. Derrode, *Représentation des formes planes à niveaux de gris par différentes approximations de Fourier Mellin analytique en vue d'indexation de bases d'images*, **Thèse de Doctorat**, Université de Rennes I, 1999.
- [Desachy 1998] J. Desachy, P. Matsakis, W. Wendling, *Représentation de la position relative d'objets 2D au moyen d'un histogramme de forces*, **Traitement du Signal (TS)**, Vol. 15, N°1, pp. 63-76, 1998.
- [Desachy 1999] J. Desachy, V. Bessettes, P. Dhérété. *Remote sensed images processing and geographic features*, dans **Information Processing for Remote Sensing**. Edité par C. H. Chen, World Scientific Publishing Co., , Chapitre 19- 25, 1999.
- [Deseilligny 1994] M. P. Deseilligny, *Lecture Automatique de Cartes*, **Thèse de Doctorat** de l'Université de Rouen, 1994.
- [Deseilligny 1995a] M. P. Deseilligny, H. L. Men, G. Stamon, *Lecture automatique des ecritures sur cartes scannées*, **Traitement du Signal (TS)**, Vol 12, N°6, pp. 639-651, 1995.
- [Deseilligny 1995b] M. P. Deseilligny, H. L. Men, G. Stamon, *Character string recognition on maps, a rotation-invariant recognition Method*, **Pattern Recognition Letters (PRL)**, Vol. 16, pp. 1297-1310, 1995.
- [Devaux 1999] P. M. Devaux, D. B. Lysak, R. Kasturi, *A complete system for the intelligent interpretation of engineering drawings*, **International Journal on Document Analysis and Recognition**



- (**IJDAR**), N°2, pp. 120-131, 1999.
- [Dhéréte 1999] P. Dhéréte, *Coopération et fusion d'opérateurs : Application au recalage automatique d'objets cartographiques*, **Thèse de doctorat** de l'université Toulouse-III, 1999.
- [Di Gesù 1999] V. Di Gesù, V. Starovoitov, *Distance-based functions for image comparison*, **Pattern Recognition Letters (PRL)**, Vol. 20., pp. 207-214, 1999.
- [Dosch 2000] P. Dosch, *Un environnement pour la reconstruction 3D d'édifices à partir de plans d'architecte*, **Thèse de Doctorat** de l'Université Nancy 1, 2000.
- [Draper 1989] B. A. Draper, R. Collins, J. Brolio, A. Hansen, E. Riseman, *The SCHEMA system*, **International Journal on Computer Vision**, Vol. 2, pp. 209-250, 1989.
- [Dudani 1977] S. Dudani, K. Breeding, R. McGhee, *Aircraft identification by moment invariants*, **IEEE Transactions on Computers**, C-26, pp39-45, 1977.
- [Eallet 2000] F. Eallet, B. Collin, G. Sella, C. Garbay, *Multi-agent architecture for scene interpretation*, **Aerospace Enhanced and synthetic vision 2000**, Orlando, Floride, USA, 2000.
- [El-Khaly 1990] El-Khaly, F. and M.A. Sid-Ahmed, *Machine recognition of optically captured machine printed Arabiv text*, **Pattern recognition**, Vol. 23, N° 11, pp. 1207-1214.
- [Ferber 1997a] J. Ferber, *Les systèmes multi-agents – vers une intelligence collective*, **éditions InterEditions**, 1997.
- [Ferber 1997b] J. Ferber, *Les systèmes multi-agents : un aspect général*, **Technique et Science Informatiques (TSI)**, Vol 16, N°8, p 979-1012, 1997.
- [Ficet 2000] V. Ficet-Cauchard, M. Revenu., C. Porquet, R. Clouard., *Aide à la conception d'applications de Traitement d'Images : une approche fondée sur le raisonnement à partir de cas*, dans **Ingénierie des Connaissances, évolutions récentes et nouveaux défis**. Edité par J. Charlet, M. Zaclad, G. Kassel & D. Bourigault, chapitre 6, éd. Eyrolles, Paris, pp. 93-108, 2000.
- [Finin 1993] T. Finin, J. Weber, *Draft specification of the KQML agent-communication language*, **The DARPA Knowledge Sharing Initiative External Interfaces Working Group**, Juin 1993.
- [Fletcher 1988] L. A. Fletcher, R. Kasturi, *A robust algorithm for text string separation from mixed text / graphics images*, **IEEE Transactionss on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 10, N°6, pp. 910-918, 1988.
- [Flusser 2000] Jan Flusser, *On the independance of rotation moment invariants*, **Pattern Recognition (PR)**, Vol. 33, pp. 1405-1410, 2000.
- [Gader 1991] P. Gader, B. Forester, M. Ganzberger, A. Gillies, B. Mitchell, M. Whalen, T. Yocum, *Recognition of handwritten digits using templates and model matching*, **Pattern Recognition (PR)**, Vol. 24, N°5, pp. 421-431, 1991.
- [Garbay 1998] C. Garbay, H. Maitre, *Interaction systèmes / environnement pour l'interprétation des signaux et des images*, **Rapport de synthèse de l'action Inter PRC 10.2 du PRC ISIS & CHM**, 1998.
- [Garcia 2001] P. Garcia, B. Couasnon, *Using a generic document recognition method for mathematical formulae recognition*, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 341-348, 2001.
- [Gardes 2001] J. Gardes, J. M. Ogier, S. Adam, R. Mullet, *CAATI, A system based dynamic document interpretation device*, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 301-311, 2001.
- [Garnesson 1991] P. Garnesson, *MESSIE I : Un Système d'Analyse de Scènes, Application à l'imagerie aérienne*, **Thèse de doctorat** de l'Université de Nice Sophia Antipolis, 1991.
- [Garris 1994] M. D. Garris, *Design, collection, and analysis of handwriting sample image databases*, **Encyclopedia of Computer Science and Technology**, Vol. 31, supplement 16, Marcel Dekker, N.Y., pp. 189-213, 1994.
- [Ghorbel 1994] F. Ghorbel, *A complete invariant description for gray level images by the harmonic analysis approach*, **Pattern Recognition Letters (PRL)**, Vol. 15, pp. 1043-1051, 1994.
- [Glauberman 1956] M. H. Glauberman, *Character recognition for business machines*, **Electronics**, **Electronics**, pp. 132-136, 1956.
- [Goldberg 1989] D. E. Goldberg, *Genetic algorithms in search, optimization and machine learning*, **Addison Wesley**, 1989.
- [Gonzalez 1992] R. C. Gonzalez, R. E. Woods, *Digital image processing*, **Addison Wesley**, New York, 1992.

- [Granlund 1972] G.H. Granlund, *Fourier preprocessing for hand print character recognition*, **IEEE Transactions on Computer**, Vol. 21, pp. 195-201, 1972.
- [Grenier 2000] V. Grenier, R. Mullet, J.M. Ogier, S. Adam, J. Gardes, Y. Lecourtier, *Une architecture distribuée pour l'interprétation de documents techniques*, dans les actes de **Reconnaissance de Formes et Intelligence Artificielle (RFIA)**, Vol. 1, Paris, pp. 427-436, 2000.
- [Grenier 2001] V. Grenier, *Contribution à l'interprétation automatique de documents techniques*, **Thèse de Doctorat** de l'Université de Rouen, juin 2001.
- [Hanson 1978] A. R. Hanson, E. M. Riseman, *VISIONS, A computer system for interpreting scenes*, dans **Computer Vision Systems**, édité par A. Hanson, E. Riseman, New York, Academic, 1978.
- [Hanson 1987] A. Hanson, E. Riseman, *The VISIONS image understanding system*, dans **Advance in computer vision**, édité par C. M. Brown, 1987.
- [Hayes 1985] B. Hayes-Roth, *A blackboard architecture for control*, **Artificial Intelligence**, Vol. 26, N°3, pp. 251-321, 1985.
- [Heroux 2001] P. Heroux, *Contribution au problème de la rétro-conversion des documents structurés*, **Thèse de doctorat** de l'université de Rouen, 2000.
- [Heutte 1998] L. Heutte, T. Paquet, J. V. Moreau, Y. Lecourtier, C. Olivier, *A structural / statistical feature based vector for handwritten character recognition*, **Pattern Recognition Letters (PRL)**, Vol. 19, pp. 629-641, 1998.
- [Hu 1962] M. K. Hu, *Visual pattern recognition by moment invariants*, **IRE Transactions on Information Theory**, Vol. 8, 1962, pp. 179-187.
- [Inggs 1995] M. Inggs, A. Robinson, *Neural approaches to ship target recognition*, dans les actes de **IEEE International Radar Conference**, pp. 386-391, 1995.
- [Joseph 1992] S.H. Joseph, P. Pridmore, *Knowledge-directed interpretation of line drawing images*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 14, N°9, pp. 928-940, 1992.
- [Kanatani 1990] K. Kanatani, *Group theoretical methods in image understanding*, dans **Springer Series in Information Sciences**, No 20, , Springer Verlag, 1990.
- [Kasturi 1990] R. Kasturi, M. Travedi, *Image Analysis Applications*, édité par R. Kasturi et M. Travedi, Marcel Dekker, New York, 1990.
- [Kearns 1997]. M. Kearns, *A bound on the error of cross validation using the approximation and estimation rates, with consequence for the training set split*, **Neural Computation**, N°9, pp. 1143-1161, 1997.
- [Khalil 2000] M. I. Khalil, M. M. Bayoumi, *Invariant 2D object recognition using the wavelet modulus maxima*, **Pattern Recognition Letters (PRL)**, Vol. 21, pp. 863-872, 2000.
- [Khotanzad 1988] A. Khotanzad, J. H. Lu, *Distortion invariant character recognition by a multi-level perceptron and back propagation learning*, dans les actes de **IEEE international Neural Networks Congress**, San Diego, pp. 625-632, 1988.
- [Khotanzad 1990a] A. Khotanzad, Y. H. Hong, *Rotation invariant image recognition using features selected via a systematic method*, **Pattern Recognition (PR)**, Vol. 23, pp. 1089-1101, 1990.
- [Khotanzad 1990b] A. Khotanzad, Y. H. Hong, *Invariant image recognition by Zernike moment*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 12, N°5, pp. 489-497, 1990.
- [Kim 1999] W.-Y. Kim and Y.-S. Kim, *Robust rotation angle estimator*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 21, N°8, pp. 768-773, 1999.
- [Kimura 1991] F. Kimura, M. Shridar, *Handwritten numerical recognition based on multiple algorithms*, **Pattern Recognition (PR)**, Vol. 24, N°10, pp. 969-983, 1991.
- [Kita 1992] N. Kita, *Object locating based on concentric circular description*, dans les actes de **IEEE International Conference of Pattern Recognition (ICPR)**, la Hague, Hollande, Vol. 1, pp. 637-641, 1992.
- [Klette 1987] R. Klette, P. Zamperoni, *Measures of correspondance between binary patterns*, **Image and Computer Vision**, Vol. 5, 287-295, 1987.
- [Kohl 1987] C. A. Kohl, A. R. Hanson, *A Goal-directed intermediate level executive for image interpretation*, dans les actes de **International Joint Conference on Artificial Intelligence (IJCAI)**, pp. 811-814, 1987.

- [Kohonen 1989] T. Kohonen, *Self-Organization and associative memory*, **Springer Verlag**, 3<sup>ème</sup> Edition, 1989.
- [Kuhl 1982] F. P. Kuhl, C. R. Gardiana, *Elliptic Fourier features of a closed contour*, **Computer, Vision Graphics and Image Processing**, Vol. 13, pp. 236-258, 1990.
- [Kundu 1989] A. Kundu, Y. He, P. Bahl, *Recognition of handwritten word : first and second order hidden Markov model based approach*, **Pattern Recognition (PR)**, Vol. 22, N°3, pp. 283-297, 1989.
- [Labiche 1998] J. Labiche, J. Gardes, E. Trupin, *Cycle de vie du document versus système d'interprétation automatique*, dans les actes de **Conférence Internationale Francophone sur l'Écrit et le Document (CIFED)**, Quebec, Canada, pp. 443-452, 1998.
- [Lai 1994] C. P. Lai, R. Kasturi, *Detection of dimension sets in engineering drawings*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 16, N°8, pp. 869-885, 1992.
- [Lai 2001] J. H. Lai, P. C. Yuen, G. C. Feng, *Face recognition using holistic Fourier invariant features*, **Pattern Recognition (PR)**, Vol. 34, pp. 95-109, 2001.
- [Lamiroy 2001] B. Lamiroy, L. Najman, E. Ehrhard, C. Louis, F. Quélain, N. Rouyer, N. Zeghache, *Scan-to-XML for vector graphics : An experimental setup for intelligent browsable document generation*, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 312-325, 2001.
- [Langrana 1997] N. Langrana, Y. Chen, A. K. Das, *Feature identification from vectorized mechanical drawings*, **Computer Vision and Image Understanding (CVIU)**, Vol. 68, N°2, pp. 127-145, 1997.
- [Le Cun 1999] Y. Le Cun, P. Haffner, L. Bottou, Y. Bengio, *Object recognition with gradient based learning*, dans **Feature grouping**, édité par D. Forsyth, Springer, 1999.
- [Lecoq 2001] J. C. Lecoq, L. Najman, O. Gibot, E. Trupin, *Benchmarking commercial OCR engines for technical drawings indexing*, dans les actes de **International Conference on Document Analysis and Recognition (ICDAR)**, Seattle, USA, pp. 138-142, 2001.
- [Lee 1991] S. W. Lee, H. S. Park, Y. Y. Tang, *Translation, scale and rotation invariant recognition of Hangul characters with transformation ring projection*, dans les actes de **International Conference on Document Analysis and Recognition (ICDAR)**, Saint Malo, France, pp. 829-836, 1991.
- [Lefèvre 1994] V. Lefèvre, *Système Multi-Agents d'Aide à la Photo-Interprétation sur Architecture Multiprocesseur*, **Thèse de doctorat** de l'université Paris XI Orsay, 1994.
- [Lefevre 2000] E. Lefevre, O. Colot, P. Vannoorenberg, D. De Bruçq, *Contribution des mesures d'information à la modélisation crédibiliste de connaissances*, **Traitement du Signal (TS)**, Vol. 17, N° 2, pp. 87-97, 2000.
- [Lefrere 1993] L. Lefrère, *Contribution au développement d'outils pour l'analyse automatique de documents cartographiques*, **Thèse de doctorat** de l'université de Rouen, 1993.
- [Lematre 1999] I. Lematre, *Contribution au développement d'outils pour l'analyse automatique des plans de réseau EDF*, **Rapport de DEA** de l'université de Rouen, 1999.
- [Lenz 1989] R. Lenz, *Describing and Recognizing Group-Invariant Pattern Classes with Group-Sampling*, **Pattern Recognition Letters (PRL)**, Vol. 9, pp. 169-173, 1989.
- [Levine 1978] M. D. Levine, *A Knowledge Based Computer Vision System*, dans **Computer Vision Systems**, édité par A. R. Hanson, E. R. Riseman, pp. 335-352, 1978.
- [Levine 1981] M. D. Levine, S. I. Shaheen, *A modular computer vision system for picture segmentation and interpretation*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 3, N° 5, pp. 540-556, 1981.
- [Levine 1985] M. D. Levine, A. M. Nazif, *Rule-based image segmentation : a dynamic control strategy approach*, **Computer Vision, Graphics and Image Processing (CVGIP)**, Vol. 32, pp. 104-126, 1985.
- [Li 1992] Y. Li, *Reforming the theory of invariants moments for pattern recognition*, **Pattern Recognition (PR)**, Vol. 25, pp. 723-730, 1992.
- [Liao 1996] S. X. Liao, M. Pawlak, *On Image analysis by moments*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 18, pp. 254-266
- [Liao 1998] S. X. Liao, M. Pawlak, *On the accuracy of Zernike moments for image analysis*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 20, N°12, pp. 1358-1364, 1998.
- [Liedtke 1997] C. E. Liedtke, J. Bückner, O. Grau, S. Growe, R. Tönjes, *AIDA: A system for the knowledge*

- based interpretation of remote sensing data*, **3<sup>rd</sup> International Airborne Remote Sensing Conference**, Copenhagen, Denmark, 1997.
- [Lladós 2001] J. Lladós, E. Valveny, G. Sánchez, E. Martí, Symbol recognition, current advances and perspectives, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 109-129, 2001.
- [Lin 1987] C. H. Lin, *New forms of shape invariants from elliptic Fourier descriptors*, **Pattern Recognition (PR)**, Vol. 20, N°5, pp. 535-545, 1987.
- [Lorette 1992] G. Lorette, Y. Lecourtier, *Reconnaissance et interprétation de textes manuscrits hors-lignes : un problème d'analyse de scène ?*, dans les actes du **Colloque Nationale sur l'Écrit et le document (CNED)**, Revue Bigre No 80, pp. 109- 135.
- [Lu 1998] Z. Lu, *Detection of text regions from digital engineering drawings*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 20, N°4, pp. 431-439, 1998.
- [Mamistvalov 1998] A. G. Mamistvalov, *n-Dimensional moment invariants and conceptual mathematics theory of recognition n-dimensional solids*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 20, N°8, 1998.
- [Matsakis 1998] P. Matsakis, *Relations spatiales structurelles et interprétation d'images*, **Thèse de Doctorat** de l'université de Toulouse III, 1998.
- [Matsuyama 1985] T. Matsuyama, V. Hwang, *SIGMA : a framework for image understanding – integration of bottom-up and top-down analyses*, dans les actes de **International Joint Conference on Artificial Intelligence (IJCAI)**, pp. 908-915, 1985.
- [Matsuyama 1990] T. Matsuyama, *A knowledge based aerial image understanding system*, **Advances in Computer Vision and Machine Intelligence**, Plenum, 1990.
- [Matsuyama 1993] T. Matsuyama, *Expert systems for image Processing, analysis and recognition : declarative knowledge representation for computer vision*, **Advances in Electronics and Electron Physics**, Vol. 86, pp. 81-171, 1993.
- [McKeown 1985] D. M. McKeown, W. A. Harvey, J. McDermott, *Rule based interpretation of aerial imagery*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, pp. 105-127, 1985.
- [McKeown 1989] D. M. McKeown, Jr. Wilson, W. A. Harvey, L. E. Wixson, *Automating Knowledge Acquisition for Aerial Image Interpretation*, **Computer Vision and Graphic Image Processing (CVGIP)**, Vol. 46, pp. 37-81, 1989.
- [Messmer 1993] B. T. Messmer, H. Bunke, *A network based approach to exact and inexact graph matching*, **technical Report IAM-93-021**, Institute für Informatik und angewandte Mathematik, Universität bern, 1993.
- [Messmer 1996] B. T. Messmer, H. Bunke, *Automatic learning and recognition of graphical symbols in engineering drawings*, dans **Graphic Recognition- Methods and Applications**, édité par R. Kasturi et K. Tombre, Lecture Notes in Computer Science, Springer Verlag, 1996.
- [Minsky 1975] M. Minsky, *A Framework for Representing Knowledge*, dans **The Psychology of Computer Vision**, édité par P. H. Winston, McGraw-Hill, New York, 1975.
- [Montesinos 1990] P. Montesinos, *Une méthodologie en vue de l'interprétation automatique d'images aériennes : Application à la détection des bâtiments*, **Thèse de doctorat** de l'université de Nice Sophia Antipolis, 1990.
- [Mukundan 1995] R. Mukundan, K. R. Ramakrishnan, *Fast computation of legendre and zernike moments*, **Pattern Recognition (PR)**, Vol. 28, pp. 433-442, 1995.
- [Mulder 1987] J. A. Mulder, A. K. Mackworth, W. S. Havens, *Knowledge structuring and constraint satisfaction : the MAPSEE approach*, **rapport technique 87-21**, Departement d'informatique de l'université de Columbia, Vancouver, Canada, 1987.
- [Mullot 2000] R. Mullot, *Interprétation de documents techniques et cartographiques : Algorithmes et Systèmes*, **Habilitation à diriger des recherches** de l'université de Rouen, 2000.
- [Mundy 1993] J. Mundy, A. Zisserman, D. Forsyth, *Application of invariance in computer vision*, **Lecture Notes in Computer Science**, Vol. 14, pp. 5-24, 1995.
- [Nagao 1979] M. Nagao, T. Matsuyama, H. Mori, *Structured analysis of complex aerial photographs*, dans les actes de **International Joint Conference on Artificial Intelligence (IJCAI)**, pp. 610-616, 1979.

- [Nagao 1980] N. Nagao, T. Matsuyama, *A Structural Analysis of Complex Aerial Photographs*, Plenum, 1980.
- [Nelder 1965] J. A. Nelder and R. Mead, *A simplex method for function minimization*, **Computer Journal**, Vol. 7, pp. 308-313, 1965.
- [Newell 1982] A. Newell, *The Knowledge Level*, **Artificial Intelligence**, Vol. 18, 1982.
- [Niemann 1990] H. Niemann, A. Brietzmann, U. Ehrlich, S. Posh, R. Regel, G. Sagerer, R. Salzbrunn, G. Schukat-Talamazzini, *A knowledge based speech understanding system*, **International Journal on Pattern Recognition and Artificial Intelligence (IJPRAI)**, Vol. 7, pp. 321-350, 1988.
- [Odorico 1997] R. Odorico, *Learning vector quantization with training count (lvqtc)*, **Neural Networks**, Vol. 10, N°6, pp. 1083-1088, 1997.
- [Ogier 1995] J. M. Ogier, R. Mullot, J. Labiche, Y. Lecourtier, *Interprétation de documents par cycles perceptifs de construction d'objets cohérents : application aux données cadastrales*, **Traitement du Signal (TS)**, Vol.12, N°6 , pp. 627-637, 1995.
- [Ogier 2000a] J. M. Ogier, R. Mullot, J. Labiche, Y. Lecourtier, *Semantic coherency : the basis of an image interpretation device – application to the cadastral map interpretation*, **IEEE Transactions on Systems, Man and Cybernetics (SMC) – part B : cybernetics**, Vol. 30, N°2, 2000.
- [Ogier 2000b] J. M. Ogier, *De l'image au document technique, problèmes d'interprétation*, **Habilitation à diriger des recherches** de l'université de Rouen, 2000.
- [Ogier 2001] J. M. Ogier, S. Adam, A. Bessaid, H. Bechar, *Automatic topographic map analysis systems*, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 229-244, 2001.
- [Ovalle 1991] A. Ovalle, C. Garbay, *KIDS: A distributed expert system for biomedical image interpretation*, dans **Information processing in Medical Imaging**, Lecture Notes in computer science, Springer Verlag, pp. 419-433, 1991.
- [Pasternak 1993] B. Pasternak, B. Neumann, *ADIK : An adaptable drawing interpretation kernel*, dans les actes de **International Joint Conference on Artificial Intelligence (IJCAI)**, Avignon, Vol. 1, pp. 531-540, 1993.
- [Pavel 1989] M. Pavel, *Fundamentals of Pattern Recognition*, **Pure and Applied Mathematics**, Marcel Dekker, Inc., New York, 1989.
- [Persoon 1977] E. Persoon, K. S. Fu, *Shape discrimination using Fourier descriptors*, **IEEE Transactions on Systems, Man and Cybernetics (SMC)**, Vol. 7, pp. 170-179, 1977.
- [Pierret 1996] C. Pierret Golbreich, *TASK, un environnement pour le developpement de systèmes à base de connaissances flexibles*, **Habilitation à Diriger des Recherches** de l'Université Paris Sud, 1996.
- [Pridmore 2001] T. P. Pridmore, A. Darwish, D. Elliman, *Interpreting Line Drawing Images : A Knowledge Level Perspective*, dans les actes de **IAPR International Workshop on Graphics REcognition (GREC)**, Kingston, Canada, pp. 349-362, 2001.
- [Prokop 1992] R. J. Prokop, A. P. Reeves, *A survey of moment-based recognition techniques for unoccluded object representation and recognition*, **Computer Vision and Graphic Image Processing (CVGIP)**, Vol. 45, N°5, pp. 438-460, 1992.
- [Ramel 1998] J.Y. Ramel, N. Vincent, H. Emptoz, *Interprétation de documents techniques par cycles perceptifs à partir d'une perception globale du document*, **Traitement du Signal (TS)**, Vol. 15, N° 1, pp. 1-20, 1998.
- [Ramel 2000] J. Y. Ramel, N. Vincent, H. Emptoz, *A structural representation for understanding line-drawing images*, **International Journal on Document Analysis and Recognition (IJ DAR)**, N°3, pp. 58-66, 2000.
- [Ramesh 1989] S. R. Ramesh, *A generalized character recognition algorithm : a graphical approach*, **Pattern Recognition (PR)**, Vol. 22, N°4, 347-350, 1989.
- [Rao 1988] A. R. Rao, A. K. Jain, *Knowledge representation and control in computer vision system*, **IEEE Expert**, pp. 64-79, 1988.
- [Rasure 1994] J. Rasure, S. Kubica, *The khoros application development environment*, dans **Experimental Environments for Computer Vision and Image Processing**, édité par H. J. Christensen, J.L. Crowley, pp. 1-32, 1994.
- [Ravichandran 1995] G. Ravichandran, M. Trivedi, *Circular Mellin features for texture segmentation*, **IEEE Transactions on Image Processing**, Vol. 4, pp. 1629-1640, 1995.

- [Resis 1991] T. H. Resis, The revised fundamental theorem of moment invariants, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 13, pp. 830-834, 1991.
- [Ribert 1998] A. Ribert, *Structuration évolutive de données : application à la construction de classifieurs distribués*, **Thèse de Doctorat** de l'Université de Rouen, 1998.
- [Ribert 1999] A. Ribert, E. Stocker, A. Ennaji, Y. Lecourtier, *Classification and feature selection by a self-organizing neural network*, dans **Lecture Notes in Computer Science**, édité par J. Mira, Springer Verlag, Berlin, pp. 365-372, 1999.
- [Rummelhart 1986] D. E. Rummelhart, J. L. Mc Clelland, *Parallel distributed processing*, Vol. 1, **MIT Press**, MA., 1986.
- [Saidali 2001] Y. Saidali, E. Trupin, J. Gardes, J. Labiche, *L'image, vecteur d'information privilégié pour le dialogue Homme-Machine-Homme*, dans les actes de la **Conférence Internationale sur le Document Electronique (CIDE)**, 2001.
- [Sandakly 1995] F. Sandakly, *Contribution à la mise en œuvre d'une architecture à base de connaissances pour l'interprétation de scènes 2D et 3D*, **thèse de doctorat** de l'université de Nice Sophia Antipolis, 1995.
- [Sekita 1988] I. Sekita, K. Toraichi, R. Mori, K. Yamamoto, H. Yamada, *Feature extraction of handwritten Japanese characters by spline function for relaxation matching*, **Pattern recognition (PR)**, Vol. 21, pp. 9-17, 1988.
- [Shen 1999] D. Shen, H. H. S. Ip, *Discriminative wavelet shape descriptors for recognition of 2-D patterns*, **Pattern Recognition (PR)**, Vol. 32, pp. 151-165, 1999.
- [Shimotsuji 1992] S. Shimotsuji, K. Maruyama, A. Matsumoto, K. Hiraki, *Agent based parallel recognition method of contour lines*, dans les actes de **International Conference of Document Analysis and Recognition (ICDAR)**, Montreal, Canada, pp. 154-157, 1995.
- [Shridhar 1984] M. Shridhar, A. badreldin, *High accuracy character recognition algorithm using Fourier and topological descriptors*, **Pattern Recognition (PR)**, Vol. 17, N°5, pp. 515-524, 1984.
- [Simard 2000] P. Y. Simard, Y. Le Cun, J. S. Denker, B. Victorri, *Transformation invariance in pattern recognition – tangent distance and tangent propagation*, **International journal of Imaging Systems and Technology**, Vol. 11, N°3, 2000.
- [Smeulders 1998] A. W. M. Smeulders, C. De Boer, *Design and performance in object recognition*, dans **Graphics recognition algorithms and systems**, édité par A.K. Chhabra et K. Tombre, Lecture Notes in computer science, Vol. 1389, Springer Verlag, pp. 335-346, 1998.
- [Sossa-Azuela 2001] J. H. Sossa-Azuela, C. Yanez-Marquez, J. L. Diaz de Leon S, *Computing geometric moments using morphological erosions*, **Pattern Recognition (PR)**, Vol. 34, pp. 271-276, 2001.
- [Tabbone 2001] S. Tabbone, L. Wendling, K. Tombre, *Indexing of technical line drawing based on F-Signature*, dans les actes de **International Conference on Document Analysis and Recognition (ICDAR)**, Seattle, USA, pp. 1220-1224, 2001.
- [Takahashi 1991] H. rakahashi, *A neural net OCR using geometrical and zonal pattern features*, dans les actes de **International Conference on Document Analysis and Recognition (ICDAR)**, Saint Malo, 1991.
- [Taxt 1990] T. Taxt, J. B. Olafsdottir, M. Daehlen, *Recognition of handwritten symbols*, **Pattern Recognition (PR)**, Vol. 23, pp. 1155-1166, 1990.
- [Taxt 1994] T. Taxt and K. Weisteen, *Classification of handwritten vector symbols using elliptic Fourier descriptors*, dans les actes de **International Conference on Pattern Recognition (ICPR)**, Jesrusalem, Israel Vol II, pp. 123-128, 1994.
- [Taza 1989] A. Taza, C. Y. Suen, *Discrimination of planar shapes, using shapes matrices*, **IEEE Transactions on Systems, Man and Cybernetics (SMC)**, SMC-19, pp. 1281-1289, 1989.
- [Teague 1980] M. Teague, *Image analysis via the general theory of moments*, **Journal of Optical Society of America (JOSA)**, 70, pp. 920-930, 1980.
- [Tombre 1996] K. Tombre, *Quelques contributions à l'interprétation de documents techniques*, **Habilitation à diriger des recherches** de l'université de Nancy I, 1996.
- [Tönjes 1999] R. Tönjes, S. Growe, J. Bückner, C.-E. Liedtke, *Knowledge based interpretation of remote sensing images using semantic nets*, **Photogrammetric Engineering and Remote Sensing**, Vol. 65, No. 7, pp. 811-821, 1999.
- [Toriu 1987] T. Toriu, H. Iwase, M. Yoshida, *An expert system for image processing*, **Fijitsu science and**

- Technical Journal**, Vol. 23, N°2, 1987.
- [Trier 1995a] O. D. Trier, T. Taxt, *Evaluation of binarization methods for document images*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 17, N°3, pp. 1191-1201, 1995.
- [Trier 1995b] O. D. Trier, A. K. Jain, *Goal-directed evaluation of binarization methods*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 17, N°2, pp. 312-315, 1995.
- [Trier 1996] O. D. Trier, A. K. Jain, T. Taxt, *Features extraction methods for character recognition – a survey*, **Pattern Recognition (PR)**, Vol. 29, pp. 641-662, 1996.
- [Trier 1997] O. D. Trier, T. Taxt, A. K. Jain, *Recognition of digits in hydrographic maps, binary versus topographic analysis*, **IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 19, N°4, pp. 399-404, 1997.
- [Unser 1996] M. Unser, *A practical guide to the implementation of the wavelet transform*, dans **Wavelets in medicine and biology**, édité par A. Aldroubi, M. Unser, CRC Press, pp. 37-73, 1996.
- [Varela 1989] F. J. Varela., *Connaître. Les sciences cognitives, tendances et perspectives*, SEUIL France, 1989.
- [Vaxivière 1994] P. Vaxivière, K. Tombre, *Knowledge organisation and interpretation process in engineering drawing interpretation*, dans les actes de **IAPR workshop on Document Analysis Systems (DAS)**, Kaiserslautern, Germany, pp. 313-321, 1994.
- [Watanabe 1990] M. Watanabe, Y. Ohta, *Operative integration of multiple stereo algorithms*, dans les actes de **International Conference on Computer Vision (ICCV)**, Osaka, Japon, pp. 476-480, 1990.
- [Weisteen 1992] K. Weisteen, T. Taxt, *Font segmentation of non rotated, printed symbols in the vector representation*, dans les actes de **IEEE. International Conference of Pattern Recognition (ICPR)**, La Hague, Hollande, pp. 669-672, 1992.
- [Winston 1970] P. H. Winston, *Learning Structural Description from Examples*, **Doctoral Dissertation**, Project MAC, MIT, 1970.
- [Wood 1996] J. Wood, *Invariant Pattern recognition : A Review*, **Pattern Recognition (PR)**, Vol. 29, pp. 1-17, 1996.
- [Wu 1986] R. H. Wu, H. Stark, *Rotation and scale invariant recognition of images*, dans les actes de **IEEE International Conference of Pattern Recognition (ICPR)**, Paris, France, 1986.
- [Wu 2001] C. H. Wu, S-J Horn, P-Z Lee, *A new computation of shape moments via quadtree decomposition*, **Pattern Recognition (PR)**, Vol. 34, pp. 1319-1330, 2001.
- [Yang 1987] H. Yang, C. C. Guest, *Performance of backpropagation for rotation invariant pattern recognition*, **International Joint Conference on Neural Networks**, San Diego, USA, Vol. IV, pp. 365-370, 1987.
- [Yu 1997] Y. Yu, A. Samal, S. Seth, *A system for recognizing a large class of engineering drawings*, dans **IEEE transactions on Pattern Analysis and Machine Intelligence (PAMI)**, Vol. 19, N° 8, pp. 868-890, 1997.
- [Yu 2001] D. Yu, H. Yan, *Separation of touching handwritten multi-numeral strings based on morphological structural features*, **Pattern Recognition (PR)**, Vol. 34, pp. 587-599, 2001.
- [Zahn 1972] C. T. Zahn, R. Z. Roskies, *Fourier descriptors for plane closed curves*, **IEEE Transactions on computers**, Vol. C. 21, N°3, pp.269-281, 1984
- [Zernike 1938] F. Zernike, *Physica I*, pp. 689, 1938.
- [Zhenjiang 2000] M. Zhenjiang, *Zernike moment-based image shape analysis and its application*, **Pattern Recognition Letters (PRL)**, Vol. 21, pp. 169-177, 2000.

---

# Annexe A : principes des algorithmes d'optimisation utilisés

## Annexe A. Principes des algorithmes d'optimisation utilisés

Dans le manuscrit de thèse, nous avons évoqué à deux reprises l'utilisation de méthodes d'optimisation. Cette annexe a pour but de fournir au lecteur un complément d'information concernant les deux outils que nous avons utilisés.

### A.1. Principes de la méthode du simplexe

La méthode du Simplex est une méthode permettant la minimisation d'une fonction à plusieurs variables. Cette méthode a été conçue par Nelder et Mead en 1965 [Nelder 1965]. Contrairement à la plupart des méthodes concurrentes, elle est basée sur une optimisation commune suivant les différentes variables, sans avoir recours à une optimisation séparée. Cependant, ce n'est pas une méthode vraiment efficace en terme de nombre d'évaluations de la fonction. Un Simplex est une figure géométrique constituée de  $N+1$  points si l'espace est à  $N$  dimensions, chacun de ces points étant connecté à tous les autres (triangle si la dimension est 2 par exemple). Si l'un des points d'un Simplex est pris comme origine, alors les  $N$  autres points constituent les axes de l'espace de représentation. La méthode du Simplex démarre avec  $N+1$  points d'origine. Pour construire ces  $N+1$  points, il est recommandé de choisir un point initial  $P_0$  puis de répartir les autres  $N$  points suivant :

$$P_i = P_0 + \lambda e_i \quad (108)$$

A partir de cette initialisation, le Simplex effectue une série d'étapes permettant de se diriger vers le point du Simplex où la valeur de la fonction est la plus grande. Ces étapes sont appelées réflexions, expansions et contractions, elles sont construites de façon à garder constant le volume du Simplex. Si c'est possible, la méthode étend le Simplex dans une direction pour faire le plus grand pas possible.



Quand une vallée est atteinte, le Simplex est contracté pour dépasser la vallée. C'est donc en fait par modification géométrique de la figure initiale que la méthode fonctionne. Le critère de terminaison est délicat à définir comme dans tout problème d'optimisation multi-dimensionnelle. Il est en effet impossible de définir une tolérance sur chacune des variables, puisqu'elles ne sont pas optimisées séparément. Néanmoins, puisqu'il est possible d'identifier des « pas », l'algorithme peut être arrêté quand la distance de déplacement des vecteurs est en valeur absolue inférieure à une fraction d'une tolérance. Une autre alternative consiste à exiger que la décroissance de la valeur de la fonction dans le pas terminal soit inférieure à un seuil. Notons qu'il est possible de choisir cette tolérance comme étant le degré de précision de la machine.

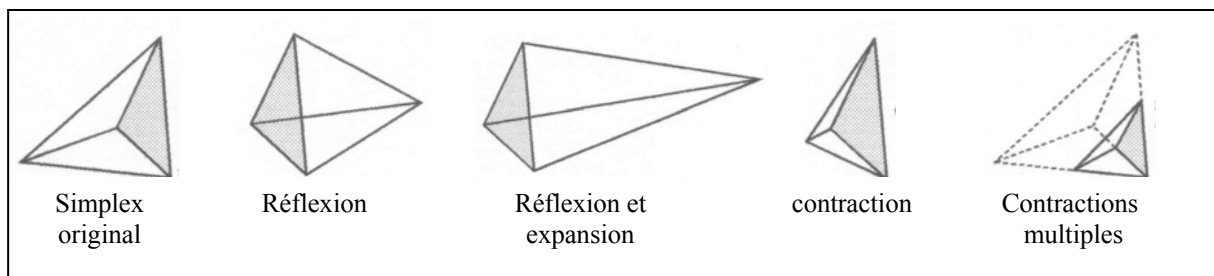


Figure A.1 : Illustration du fonctionnement de la méthode du Simplex [Nelder 1965]

Dans le cadre de notre travail de thèse, cet algorithme du Simplex a été utilisé pour la minimisation du critère défini par (94). Le but de cette optimisation est l'estimation des paramètres de mouvement des formes. Les résultats obtenus sont tout à fait corrects, surtout lorsque le Simplex est lancé deux fois consécutivement. Dans ce cadre, les résultats d'une première optimisation sont utilisés pour l'initialisation de la seconde, ce qui permet d'affiner le résultat obtenu. Toutefois, il faut signaler que la méthode est beaucoup plus lente que les Algorithmes Génétiques que nous présentons ci-après.

## **A.2. Principes des Algorithmes Génétiques (A.G.)**

De par leur nom, les algorithmes génétiques [Goldberg 1989] évoquent la biologie. Développés par John Holland en 1975, ils sont fondés sur les mécanismes de la sélection naturelle et de la génétique en s'appuyant sur la constatation que « *Les organismes vivants résolvent superbement de nombreux problèmes* ». Trois principes de base sont nécessaires à la bonne compréhension de la suite :

- Une solution possible du problème d'optimisation est représentée « chromosomiquement » par une chaîne de lettres appartenant à un alphabet qui est le plus souvent l'alphabet binaire (même si ce n'est pas une condition *sine qua non*). L'utilisation d'A.G. passe donc par un codage des paramètres du problème d'optimisation de façon binaire.
- Des moyens de créer une population initiale de solutions possibles sont nécessaires puisque les A.G. vont travailler non pas sur un unique individu en cherchant à améliorer ses performances mais sur un ensemble de solutions (individu) appelé population.

- Des opérateurs pseudo-aléatoires inspirés de la génétiques (reproduction, croisement, mutation..) permettent, au fur et à mesure de l'exécution de l'A.G., de faire évoluer la population (ensemble de solutions possibles) en assurant que les générations successives seront de plus en plus adaptées à la résolution du problème en exploitant efficacement l'information obtenue précédemment. Pour ce faire, ces opérateurs utilisent à la fois les principes de la survie des structures les mieux adaptées et les échanges d'information pseudo aléatoires, pour former un algorithme d'exploration qui possède certaines des caractéristiques de l'exploration humaine. Ce mécanisme favorise donc les éléments les meilleurs, les plus aptes.

A partir de ces principes, un AG comprend 4 étapes principales, la première est effectuée une fois, lors de l'initialisation, puis les 3 autres sont itérées jusqu'à validation d'un critère de terminaison.

- Comme nous l'avons dit dans la partie précédente, les algorithmes génétiques nécessitent le codage de l'ensemble des paramètres du problème d'optimisation en une chaîne de caractères de longueur finie. Ce codage va correspondre à une discrétisation de l'espace paramétrique (ensemble d'hypercubes). La plupart du temps, l'alphabet de caractères est binaire. Une fois le codage défini, il est possible de générer aléatoirement une population initiale dont la taille est à définir. A titre d'exemple, si la fonction à optimiser compte deux paramètres a et b dont les valeurs varient de 0 à 31 par pas de 1, chacun de ces paramètres occupera 5 bits ( $2^5=32$ ) et la population sera constituée de n individus de la forme :

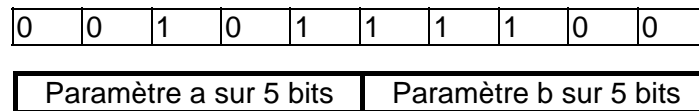


Figure A.2. : Exemples de chaîne constituant un individu ( $a=5$ ,  $b=28$ )

- La reproduction est le premier opérateur. Celui-ci consiste à copier ou non chaque chaîne de la population en fonction des valeurs de la fonction. Elle permet d'obtenir une nouvelle population contenant autant d'individus que la population initiale mais dans laquelle les individus les plus forts sont représentés alors que les individus plus faibles sont éliminés. Cet opérateur est donc une version artificielle de la sélection naturelle (la survie Darwinienne) des chaînes les plus adaptées. Deux « sous-opérations » peuvent être distinguées dans la reproduction :
  - le calcul de l'adaptation (fitness) de chacun des individus d'une population. C'est le calcul des probabilités pour chacun d'être à l'origine de la génération suivante. Du point de vue génétique, cette fonction joue le rôle de l'environnement qui va juger la « force » d'un individu. Elle permet d'attribuer à chaque individu une probabilité de se reproduire pour former la génération suivante,

en fonction de la valeur que prend la fonction à optimiser. Les contraintes que doit respecter une telle fonction nous ont amené à la définir de la façon suivante :

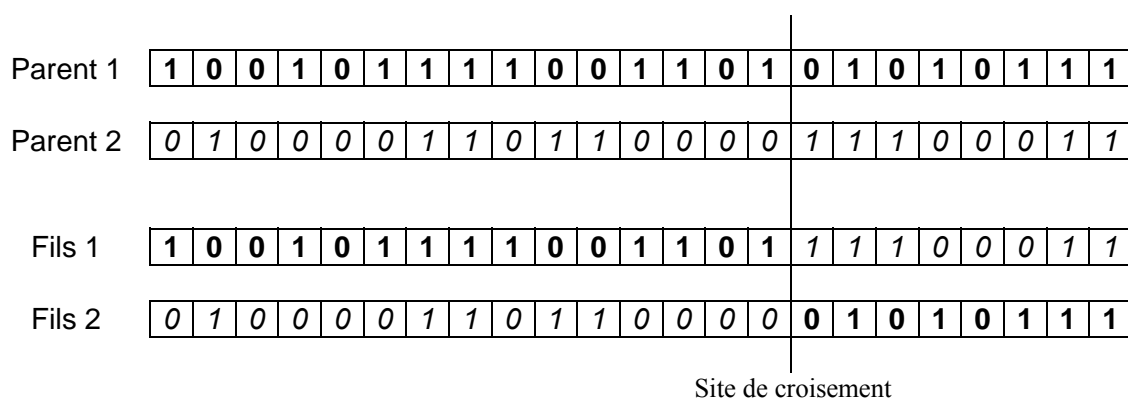
$$f(S) = \alpha (S - S_{\min} + a) \tag{109}$$

avec :

$$a = \frac{1}{1 + \sum_{Pop} S - N \times S_{\min}} \quad \text{et} \quad \alpha = \frac{1}{\sum_{Pop} S - N \times S_{\min} + N \times a} \tag{110}$$

Dans ces expressions,  $S$  est le score (la valeur prise par la fonction) de l'individu et  $S_{\min}$  est le score obtenu par le moins bon individu.

- le choix, en fonction des valeurs obtenues à l'étape précédente, des individus survivants dans la génération suivante. Ce « tirage » permet de générer une nouvelle population contenant autant d'individus que la précédente. Une des solutions algorithmiques pour effectuer ce tirage est d'assimiler celui-ci à une roue de loterie biaisée pour laquelle chaque chaîne de la population courante occupe une section de la roue proportionnelle à son adaptation :
- Le croisement est un opérateur permettant de simuler le mélange des gènes qui a lieu lors de la reproduction naturelle d'un couple d'individus. Il permet de créer de nouvelles solutions à partir des anciennes par échange de matériel génétique. Cet opérateur peut se décomposer en deux étapes. Dans un premier temps, les éléments issus de la reproduction sont appariés aléatoirement puis, chaque paire formée peut subir (nouveau tirage aléatoire guidé par une probabilité de croisement) une permutation. Cette permutation peut s'effectuer en un ou plusieurs endroits de la chaîne (les sites de croisement) qui sont eux aussi choisis aléatoirement.



*Figure A.3. : Illustration de l'opérateur de croisement*

- La mutation est un opérateur qui permet de résoudre les problèmes d'exploration de sous espaces de solutions et donc de se sortir des problèmes d'*extrema* locaux. La mutation qui consiste à changer un « 1 » en « 0 » ou inversement de façon aléatoire (une probabilité de mutation est ici utilisée) permettra de sortir de telles impasses si tous les individus convergent vers celle-ci.

Avant mutation	1	0	0	1	0	1	1	1	1	0	0	1	1	0	1	0	1	0	1	1	1	
Après mutation	1	0	0	1	0	1	0	1	1	0	0	1	1	0	1	0	1	0	1	0	1	1

Figure A.4. : Illustration de l'opérateur de mutation

A partir de ces opérateurs, la première étape nécessaire à l'exécution d'un algorithme génétique est la création d'une population initiale. Cette population est créée de manière aléatoire (pour chaque gène de chaque individu de celle-ci, un « pile ou face » est simulé). Ensuite, pour faire évoluer cette population génération après génération, chacun des opérateurs défini précédemment (reproduction, croisement et mutation) est appliqué de façon itérative aux populations successives. Ceci jusqu'à ce qu'un critère d'arrêt soit validé pour arrêter l'algorithme. Nous n'irons pas plus en avant concernant ces algorithmes génétiques, mais nous encourageons le lecteur intéressé à se reporter à l'URL <sup>13</sup> où sont disponibles en ligne des compléments d'information et la documentation de la bibliothèque correspondante [Adam 1999a].

Nous avons exploité ces algorithmes génétiques pour deux applications. La première concerne l'optimisation de (94) pour l'estimation des paramètres de mouvement. Un individu correspond dans ce cadre à un couple de valeurs  $(\alpha, \beta)$ . Les résultats obtenus sont du même ordre que ceux obtenus avec l'algorithme du Simplex mais dont le temps de calcul est plus réduit.

La seconde application concerne la combinaison de primitives, en associant un individu des A.G. à une combinaison donnée des différents invariants présentés dans le manuscrit. Nous avons déjà évoqué les résultats obtenus dans le cadre de la discussion menée en 2.4.5. Le lecteur intéressé par cette application en trouvera la description complète dans [Adam 2000b].

---

<sup>13</sup> <http://psiserver.insa-rouen.fr/psi/fr/accueil.html>



---

# Annexe B : Un modèle statistico-structurel pour la reconnaissance de symboles

## Annexe B .Un modèle statistico-structurel pour la reconnaissance de symboles

Dans cette annexe, nous présentons un outil dédié à la reconnaissance de symboles sur des documents techniques. Cet outil, basé sur une combinaison d'une approche statistique et d'une approche structurelle, et présenté dans [Adam 2000a], a été développé dans le cadre d'un stage de DEA [Adam 1998]. Récemment, ces travaux ont été poursuivis par un autre stage de DEA dans notre équipe [Delalandre 2001].

### B.1. Le problème

A diverses reprises dans le manuscrit, nous avons évoqué l'importance de la reconnaissance des symboles lors de l'interprétation d'un document technique, quel qu'en soit le type. Dans ce contexte, pour notre première application visant à l'interprétation des plans de réseaux France Telecom, le cahier des charges de *NATALI* nous imposait la reconnaissance de deux type de symboles particuliers :

- les chambres, dont la représentation est relativement homogène sur les différents plans (le symbole de droite sur la figure C.1.).
- les **Points de Concentration**, dont de nombreuses variantes existent (les deux symboles de gauche sur la figure C.1.).

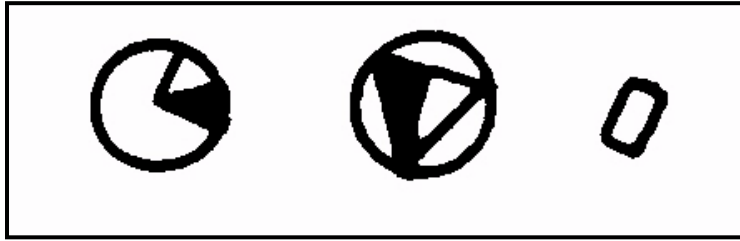


Figure B.1 : Les symboles à reconnaître sur les documents réseaux France Telecom

## B.2. L'approche choisie

Les figures C.1. et C.2. montrent que les symboles à reconnaître sont caractérisés par un agencement particulier d'occlusions. Dans un tel cas de figure, lorsque des formes peuvent être définies par association de formes élémentaires, un modèle structurel de représentation des formes est bien adapté. C'est donc vers ce type d'approches que nous nous sommes orienté. Néanmoins, la figure C.2. montre que les occlusions à prendre en compte sont de formes différentes. Cela impose, dans un premier temps de reconnaître ces occlusions. Nous avons opté pour une reconnaissance statistique de ces occlusions, basée sur les modèles développés dans la partie 1. C'est pourquoi nous appelons notre modèle pour reconnaître ces symboles statistico-structurel. Cette extraction de modèle statistico-structurel se déroule en trois étapes successives :

- Extraction des occlusions présentes dans l'image, cette étape est illustrée par la figure C.2.

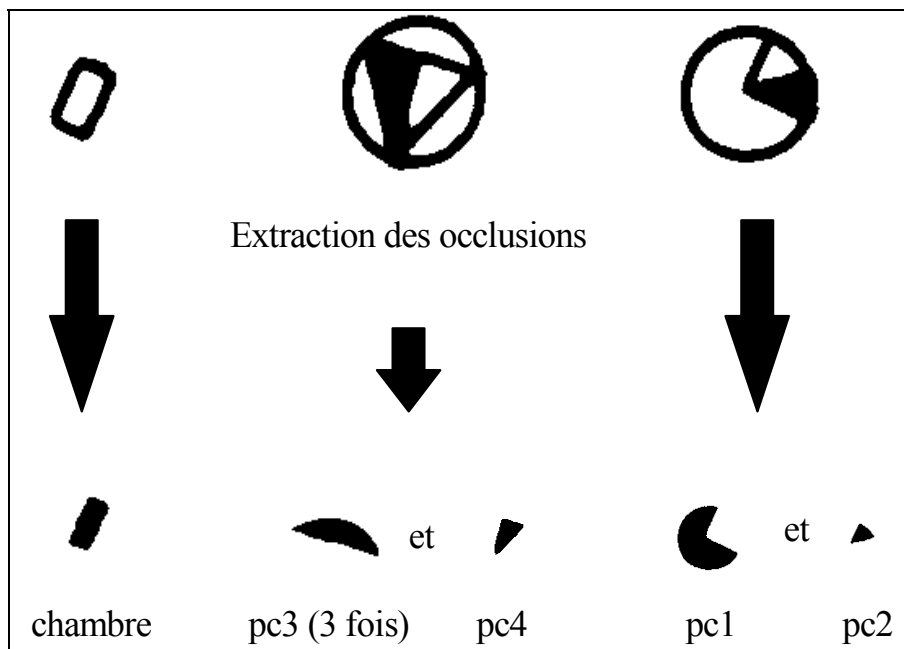


Figure B.2 : L'extraction des occlusions constituant les symboles

- Reconnaissance des occlusions extraites. Pour cette étape, nous utilisons les invariants de Fourier Mellin présentés dans la première partie du document que nous combinons avec des descripteurs

plus intuitifs tels que la surface, le périmètre ainsi que le nombre de vecteurs issus de la polygonalisation des contours des occlusions. Ce modèle est ensuite associé à un classifieur de type **k Plus Proche Voisin**. La constitution de la base, ainsi que les résultats obtenus pour cette phase de reconnaissance des occlusions sont présentés dans la matrice de confusion C.3. Cette matrice correspond à un taux de bonne reconnaissance de 91,38%, taux qui pourrait probablement être amélioré en augmentant la taille de la base qui n'est ici composée que de 64 symboles.

Classe	Cardinalité
Pc1	4
Pc2	4
Pc3	42
Pc4	14
Chambre	25
Bruit	21

	PC1	PC2	PC3	PC4	C	B
PC1	100%	0%	0%	0%	0%	0%
PC2	0%	100%	0%	0%	0%	0%
PC3	0%	0%	90%	0%	0%	10%
PC4	0%	0%	0%	100%	0%	0%
C	0%	0%	0%	0%	92%	8%
B	0%	0%	0%	8%	15%	77%

*Figure B.3 : Composition de la base d'occlusions et résultats obtenus*

A la suite de cette étape nous disposons de l'ensemble des occlusions contenues dans l'image, qui ont été étiquetées grâce au mécanisme décrit ci-avant.

- L'étape suivante consiste alors à analyser l'agencement des occlusions reconnues afin de les associer en symbole. Dans [Adam 1998], l'approche adoptée consiste à parcourir l'image avec une fenêtre et à rechercher la présence des occlusions correspondant à un symbole PC, en examinant la liste des hypothèses issues de la classification. Lorsqu'une configuration correspondant aux modèles de symboles est obtenue, un objet « PC » est construit et les occlusions correspondantes sont supprimées de la liste.

### **B.3. Les résultats obtenus**

Cette approche a permis l'obtention de bons taux de reconnaissances puisque 90% des PC et des chambres sont reconnus. Notons que ceci nous permet d'effacer les symboles qui sont connectés au réseau de l'image, afin d'ensuite correctement vectoriser ce réseau (Figure C.4). De plus, cette approche a surtout montré la faisabilité d'une combinaison statistique / structurel. Toutefois, il est important de préciser que ces tests ont été effectués sur de très petites bases, et avec un nombre de classes très restreint.



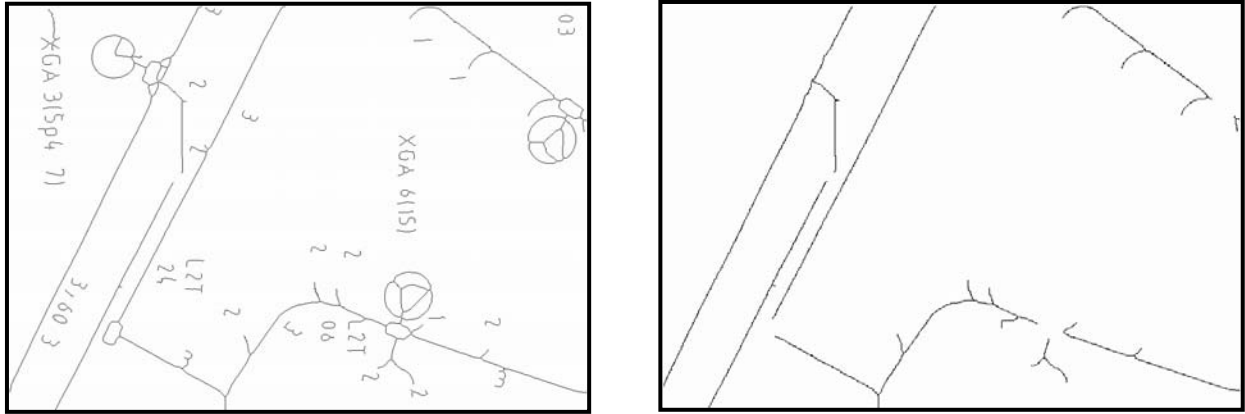


Figure B.4 : Illustration de l'intérêt de la suppression des symboles avant vectorisation

Avec cette approche, il est également important de noter que l'ajout d'un nouveau symboles exige, dans ces premiers travaux, une importante modification du code dans lequel la connaissance est intégrée. C'est pourquoi nos travaux actuels, dans le cadre de [Delalandre 2001], visent à généraliser et à formaliser notre approche combinant statistique et structurel. Dans ce contexte, à la suite de la reconnaissance des occlusions, un outil de génération de graphes d'occlusions est utilisé. Cet outil permet la génération de graphes sous contraintes. Il est ainsi possible de limiter le nombre de connexions autorisées, de borner les distances entre occlusions acceptées pour la création d'arc, ou de générer un graphe totalement interconnecté. Les graphes sont décrits en XML<sup>14</sup>. L'utilisation de ce langage nous apporte une généricité importante et sa facilité de manipulation par l'intermédiaire d'un processeur XSLT<sup>15</sup> nous permet de générer différents modèles de graphes. De plus, ce processeur autorise une transformation aisée des graphes en un format SVG, ce qui permet une visualisation des résultats (Figure C.4)

Notons à ce titre que XML et ses sous langages (SVG<sup>16</sup>, XGMML<sup>17</sup>...) tendent à devenir des standards pour la représentation des données dans notre équipe. A titre d'exemple, nous avons implanté, en utilisant le même formalisme de représentation que pour les occlusions, un outil de démonstration permettant de reconnaître des formes élémentaire (carrés, rectangles...) à partir des résultats SVG d'une vectorisation. Dans le même temps, une implémentation de XGMML est en cours pour la représentation de nos bases de graphes.

La figure C.5. illustre l'utilisation de XSLT afin de construire une représentation graphique de notre base de graphes, initialement représentés en XML à la suite de la reconnaissance des occlusions et de la construction d'un graphe sous contraintes.

<sup>14</sup> eXtensible Markup Language

<sup>15</sup> eXtensible Stylesheet Language Transformation

<sup>16</sup> Scalable Vectoriel Graphic

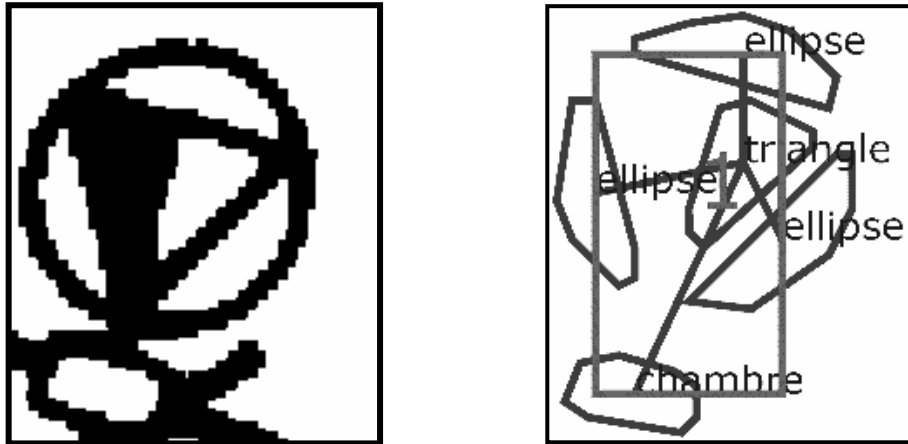


Figure B.5 : Image de PC et reconstruction SVG du graphe d'occlusions correspondant

La dernière étape du processus de reconnaissance des symboles consiste à appairer les graphes candidats avec des graphes modèles. Pour ce faire, nous utilisons un outil de recherche d'isomorphisme développé au sein du laboratoire [Heroux 2001]. Cet outil utilise un calcul de distance « paramétrable » basé sur les taux de recouvrement entre le graphe candidat, le graphe modèle et leur sous-graphe commun. Il est possible de choisir si seule la topologie du graphe doit être utilisée, ou si les étiquettes, tant des arcs que des nœuds doivent être prises en compte. Dans ce contexte, nous envisageons actuellement la mise en œuvre d'une distance entre nœud tenant compte de la matrice de confusion présentée en Figure C.3. Dans cette perspective, nous avons donc explicité la connaissance de notre outil, contrairement à la première implémentation [Adam 1998]. Il est désormais possible d'étendre notre base de modèles sans modification des traitements. Ce travail ouvre donc des perspectives très encourageantes, parmi lesquelles des travaux sur le choix de la meilleure modélisation sous forme de graphes, ou la combinaison de telles approches avec les approches évoquées dans le manuscrit.

---

<sup>17</sup> eXtensible Graph Markup and Modeling Language



---

# Annexe C : Un cas d'usage de *NATALI v2*

## Annexe C .Un cas d'usage de *NATALI V2*

Dans le corps du manuscrit, deux principales contributions ont été proposées ; la première concerne des outils de reconnaissance de formes multi-orientées et multi-échelles ; la seconde aborde l'intégration de ces outils dans un système d'interprétation de documents techniques à base de connaissances. Dans cette annexe, nous proposons de montrer un exemple d'utilisation du système *NATALI v2* illustrant ces deux contributions ainsi que les perspectives évoquées en fin de manuscrit.

### C.1. Le problème considéré

#### C.1.1. L'image traitée

L'image que nous proposons de traiter (Figure C.1) est issue d'un plan A0 de Toulouse (site pilote dans le cadre du projet *NATALI*). La planche papier a fait l'objet d'une numérisation en 256 niveaux de gris, à une résolution de 400 DPI. Elle a ensuite été découpée en tuiles de 2000\*2000 pixels.

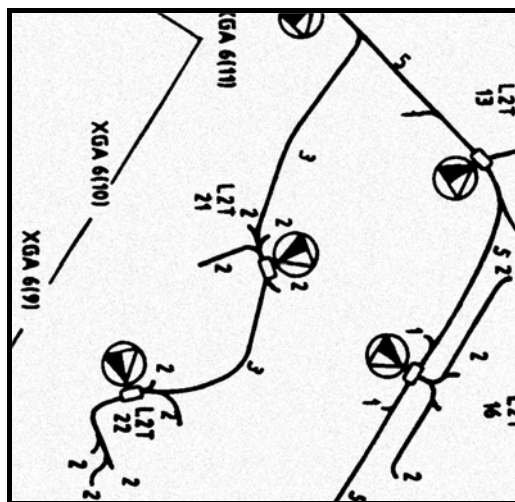


Figure C.1 : l'image traitée dans l'exemple d'exécution

Cette image contient, au même titre que la plupart des plans de réseau France Telecom, 3 principaux types d'objets du domaine :

- ❑ le tracé des câbles, c'est à dire l'ensemble des Tronçons d'ARtère (**TAR**) du plan, qui peuvent être associés à des caractères qui précisent les propriétés des câbles ;
- ❑ les Points de Concentration (**PC**), qui sont constitués d'un symbole et d'une chaîne de caractères de type « L2T + Numéro » ;
- ❑ les chambres, qui sont constituées d'un symbole et d'une chaîne de caractères de type « XGA 6 (Numéro) ».

### C.1.2. Les intentions d'interprétation

Le système présenté dans la seconde partie du manuscrit est en mesure d'extraire les instances des différents types d'objets présents dans le réseau représentant les connaissances du domaine. Dans le cas présent (qui correspond au modèle présenté en 3.5.2.), il est possible d'exprimer des intentions d'extraction concernant les objets « Réseau », « Tronçons d'ARtère », « Point de Concentration » et « Chambre ». Cette intention est émise par l'intermédiaire de l'interface proposée sur la Figure C.2. Les différents types d'objets sont proposés dans le menu « receiver » de l'interface, qui se base sur le contenu du réseau sémantique. Le choix de l'image et l'attribution d'un identifiant à celle-ci sont effectués dans le menu *Contenu*. Nous prenons ici l'exemple d'une intention concernant l'extraction des Points de Concentration présents sur l'image pour montrer le cheminement de l'interprétation.

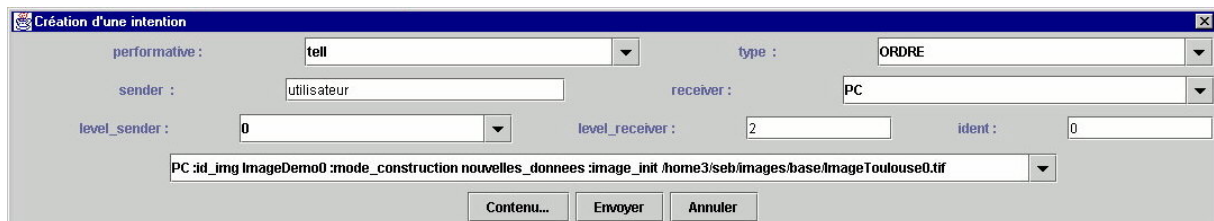


Figure C.2 : Interface d'émission d'une intention d'interprétation

## C.2. Déroulement de l'interprétation

Le système utilisé dans le cadre de cet exemple est présenté sur la figure C.3. Les flèches symbolisent les relations de constitution présentes dans le réseau sémantique et qui permettent donc la structuration en couches du système. Notons que ce dernier intègre, contrairement à *NATALI v1*, des possibilités de compétition (Figure 3.25 et 3.26) puisque trois agents permettent la construction de caractères. Ce point va nous permettre d'illustrer l'intérêt de l'acquisition et de la représentation des connaissances graphiques (stratégiques et descriptives) ainsi que l'apport du système multi-agents développé.

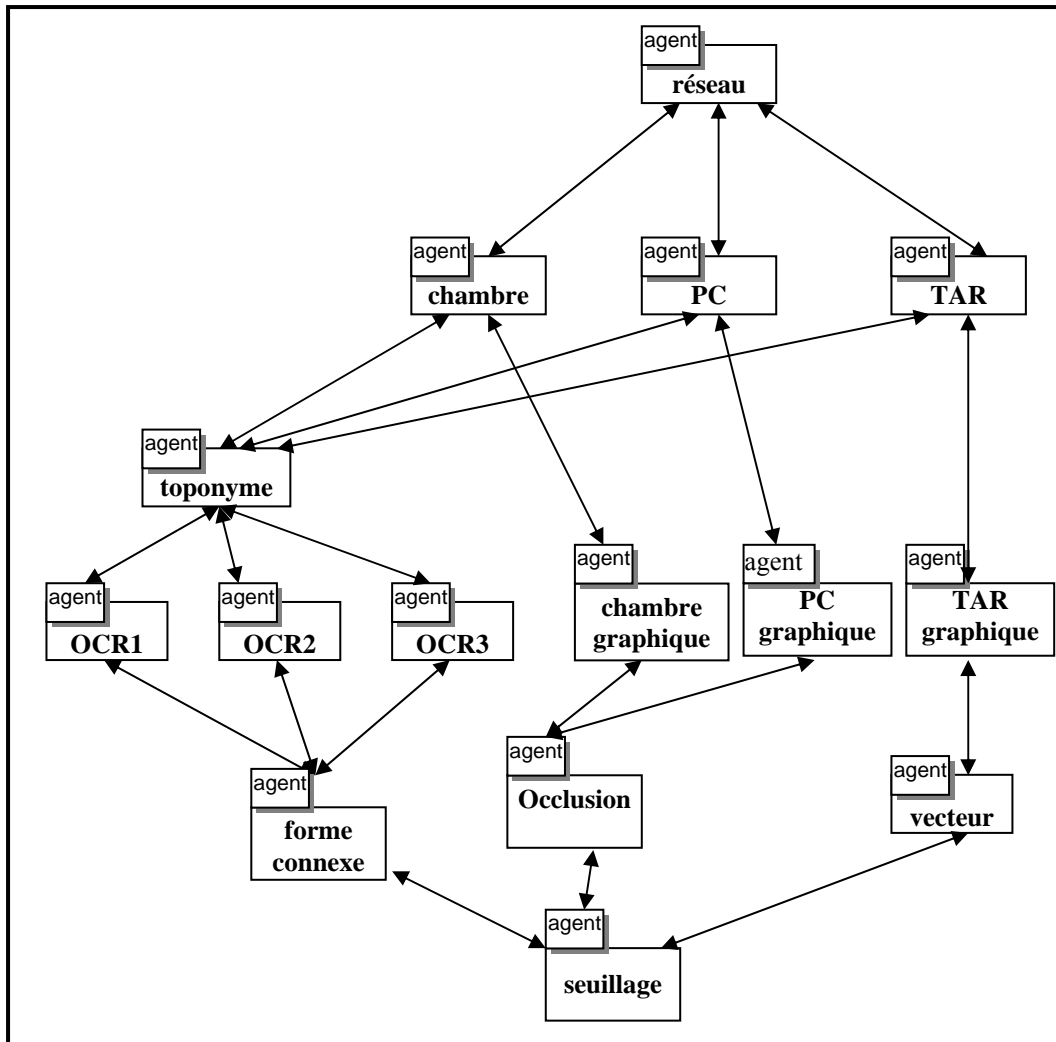


Figure C.3 : le système multi-agents utilisé dans le cadre de l'exemple

L'interprétation se déroule en deux phases successives.

### C.2.1. La phase de propagation de l'intention

L'intention émise par l'intermédiaire de l'interface présentée en Figure C.2 est d'abord envoyée à l'agent PC dans le système présenté sur la figure C.3. Le message est transmis à cet agent grâce à son nom symbolique qui correspond au nom de la donnée à construire. Par consultation du réseau sémantique explicitant le modèle des connaissances descriptives, l'agent « PC » sait alors que l'objet « PC » est constitué d'un symbole (le PC\_graphique) et d'un toponyme. Il sait également que ces deux objets sont associés à la fois par une relation de voisinage (**V**), dont il connaît les caractéristiques, et par une relation de caractérisation (**C**) correspondant à la grammaire des toponymes associés aux « PC ». Comme nous l'avons précisé en 3.5.2.1, la relation de voisinage est utilisée pour la construction des données alors que la relation de caractérisation associée à la grammaire est utilisée pour la validation de la cohérence des objets reconstruits.

Compte tenu des relations de décomposition présentes dans le réseau sémantique, l'agent PC va déléguer aux agents « PC\_graphiques » et « toponymes » la construction des données correspondantes. Le protocole des réseaux contractuels est utilisé pour cette délégation. Toutefois, notre système intégrant dans sa version actuelle une seule méthode de construction de chacune de ces données, il n'y a pas réellement de négociation. Les agents « toponymes » et l'agent « PC\_graphique » vont être les deux seuls à répondre à la requête émise par l'agent « PC ». Dans cette réponse, ils vont stipuler qu'ils sont capables de fournir à l'agent « PC » les données qu'il a demandées. L'agent « PC » va alors leur adresser un « ordre de construction ».

A la réception de cet ordre, les deux agents vont à leur tour consulter leur base de connaissances descriptives.

- L'agent PC\_graphique correspond à un « objet » de type **A** (Symbole) présentant la particularité de pouvoir être connecté à d'autres éléments graphiques (ce qui est explicité dans le réseau sémantique par la propriété de voisinage). Dans ce cas, les approches à base d'invariants ne peuvent actuellement pas être utilisées pour les raisons que nous avons données dans la conclusion générale du mémoire. Le scénario de reconnaissance de symboles qui va être déclenché est celui qui est présenté en annexe B. Il permet en effet de s'affranchir des problèmes de connexions. Il exploite une représentation structurelle des formes basée sur les occlusions. De ce fait, l'agent PC\_graphique va demander à un agent chargé de l'extraction des occlusions de lui fournir les instances de celles-ci. Notons que ceci implique le déclenchement d'une binarisation qui est effectuée par l'agent image binaire, à la demande de l'agent extrayant les occlusions.
  
- L'agent « toponyme » va quant à lui exploiter les compétences des agents OCR qui ont la capacité de construire des caractères. Dans ce cas, puisque nous disposons de trois agents capables d'effectuer une telle tâche, le protocole du réseau contractuel est vraiment mis en œuvre. Les scores de compétence des différents agents sont exploités (les connaissances stratégiques graphiques du système). Dans le cas du texte, nous aurions pu utiliser les résultats obtenus sur de grandes bases de caractères pour avoir des scores statistiquement représentatifs (cf. 2.4). Toutefois, pour montrer l'intérêt de la phase d'acquisition des connaissances graphiques, les scores utilisés dans le cadre de cet exemple sont issus d'un apprentissage exploitant l'interface présentée en Figure 3.27. La moyenne pondérée des taux de reconnaissance par classe est utilisée comme score d'aptitude pour chacun des agents OCR.

### **C.2.2. La phase de remontée des informations**

Sur la figure C.4 nous montrons les résultats obtenus lors de la phase de construction des PC\_graphique. Toutes les occurrences de PC\_graphiques complets ont été correctement détectées.

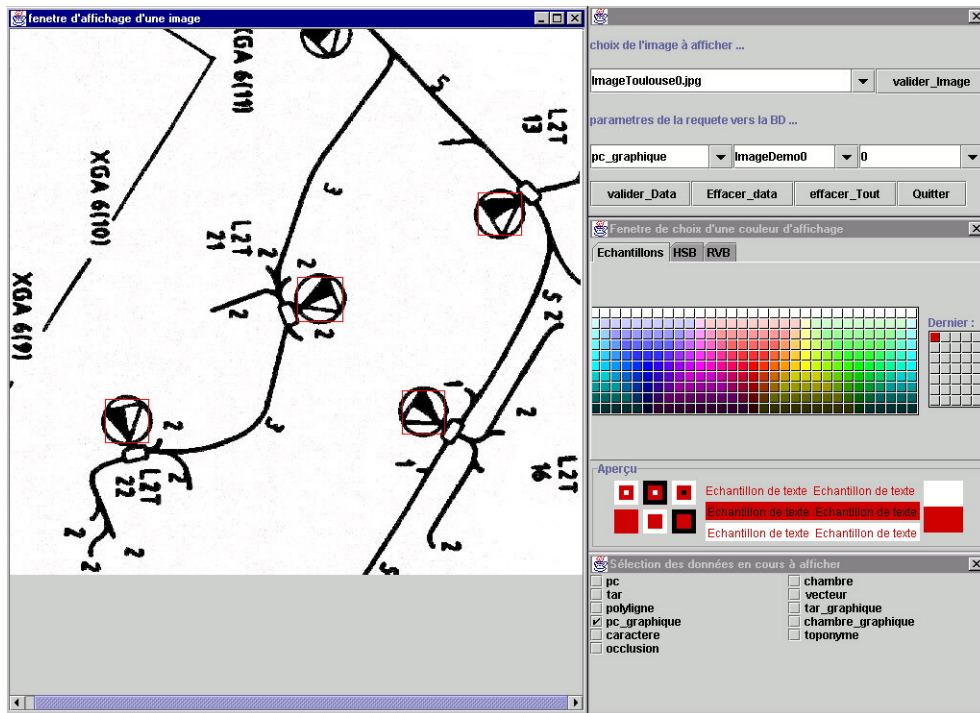


Figure C.4 : Les résultats obtenus par l'agent de reconnaissance des symboles PC

Sur la Figure C.5, nous présentons les résultats obtenus par les agents chargés de la reconnaissance de caractères. Ces agents utilisent les trois approches présentées dans la partie 2 du manuscrit. Ils prennent en entrée la même image binaire que dans le cas des symboles. Un filtrage des formes connexes est effectué à l'aide de seuils qui sont issus d'un apprentissage effectué sur une autre tuile issue du même plan. L'image issue de ce filtrage ne contient que des formes connexes dont la taille est compatible avec celle des caractères. Le scénario de reconnaissance de la figure 2.3 est ensuite lancé. Les bases d'apprentissage utilisées pour la reconnaissance sont issues de la même tuile que celle qui a permis l'apprentissage des seuils de taille des formes connexes. Elles ont été obtenues par utilisation de l'interface présentée en Figure 3.27.

Concernant les connaissances stratégiques de l'agent « toponyme », nous avons opté pour la stratégie de déclenchement de « tous les compétents ». Tous les agents capables de reconnaître des caractères sont donc exécutés et nous obtenons pour chaque forme connexe, trois hypothèses de reconnaissance (Figure C.5). Chacune des hypothèses est stockée dans la base de données et peut être identifiée par le scénario qui a permis de l'obtenir. Quand chacun des agents ayant répondu positivement à l'appel d'offre lancé par l'agent toponyme a fini la construction, un message de fin de travail est envoyé à l'agent toponyme.



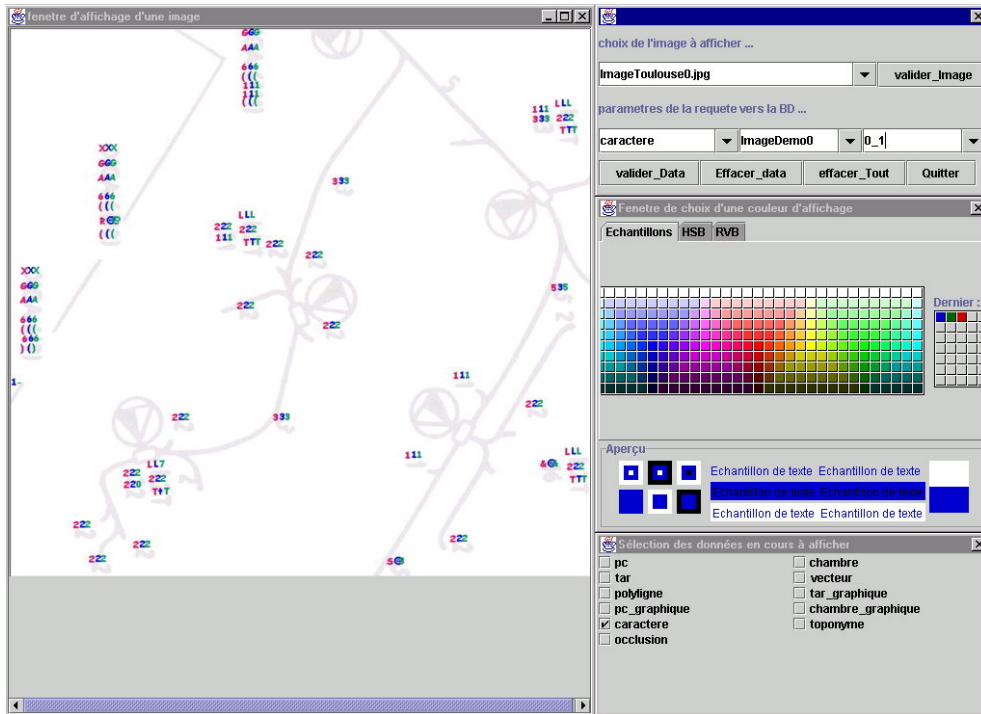


Figure C.5 : Les résultats obtenus par les agents de reconnaissance de caractères

La figure C.6 présente les résultats obtenus par l'extraction des toponymes de l'image, pour chacun des *scenarii* de reconnaissance de caractères qui a été déclenché. La méthodologie adoptée pour cette construction est équivalente à celle qui est proposée dans [Fletcher 1988]

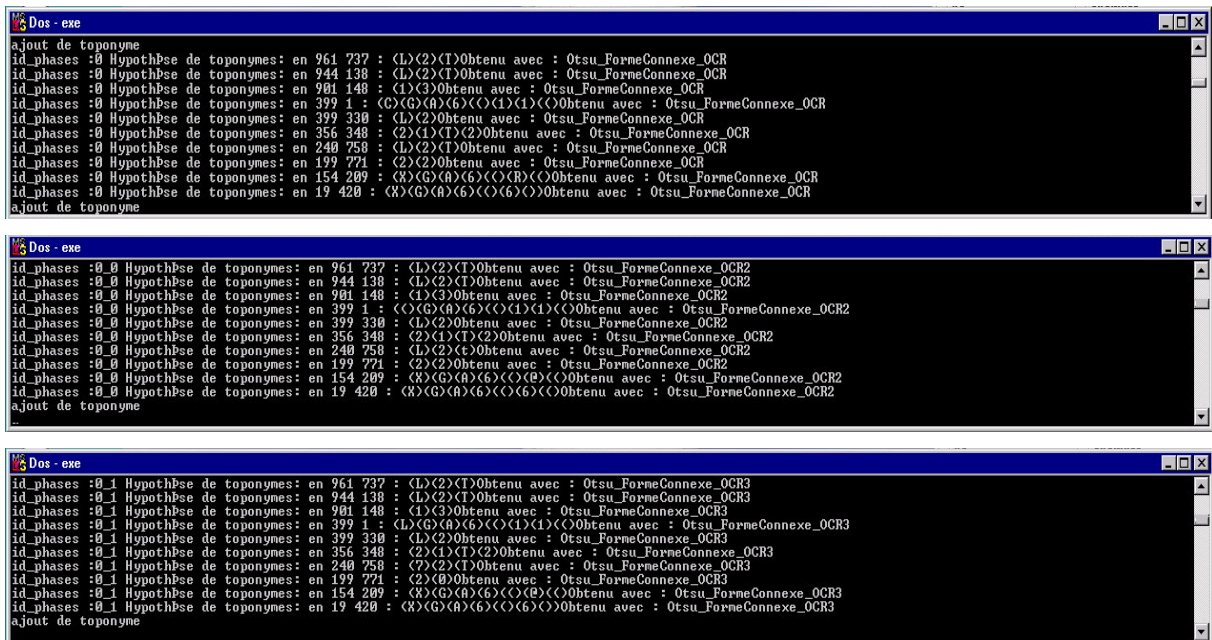


Figure C.6 : Les résultats obtenus par les agents de construction des toponymes

Sur cette figure C.6, nous mettons en évidence le fait que la phase de remontée des informations permette de stocker, au fur et à mesure de la reconstruction d'information, le scénario qui a généré les

données. Cela permettra, comme nous l'avons précisé en 3.5.3.3, l'évaluation de la qualité de la chaîne déclenchée lorsque les données du niveau « métier » auront été reconstruites.

Lorsque le stockage de l'ensemble des données a eu lieu, les agents PC\_graphique et toponymes envoient un message à l'agent PC pour lui stipuler que les données sont disponibles en lecture dans la base de données. Celui-ci peut alors commencer la phase de construction qui exploite la relation de voisinage présente dans le réseau sémantique. C'est à ce niveau qu'interviennent les constructeurs génériques évoqués dans la partie du manuscrit concernant les perspectives. L'agent émet ainsi des hypothèses de présence de Points de Concentration qui sont associées aux différents scénarii déclenchés. La proximité entre les 4 coins du rectangle englobant le symbole et les 4 coins du rectangle englobant le toponyme reconstruit est utilisée pour ce faire. Si l'une des distances calculées est comprise dans l'intervalle défini dans le réseau sémantique, une hypothèse de PC est émise. En utilisant ce principe, 12 hypothèses de PC complets sont générées, elles sont montrées sur la figure C.7.

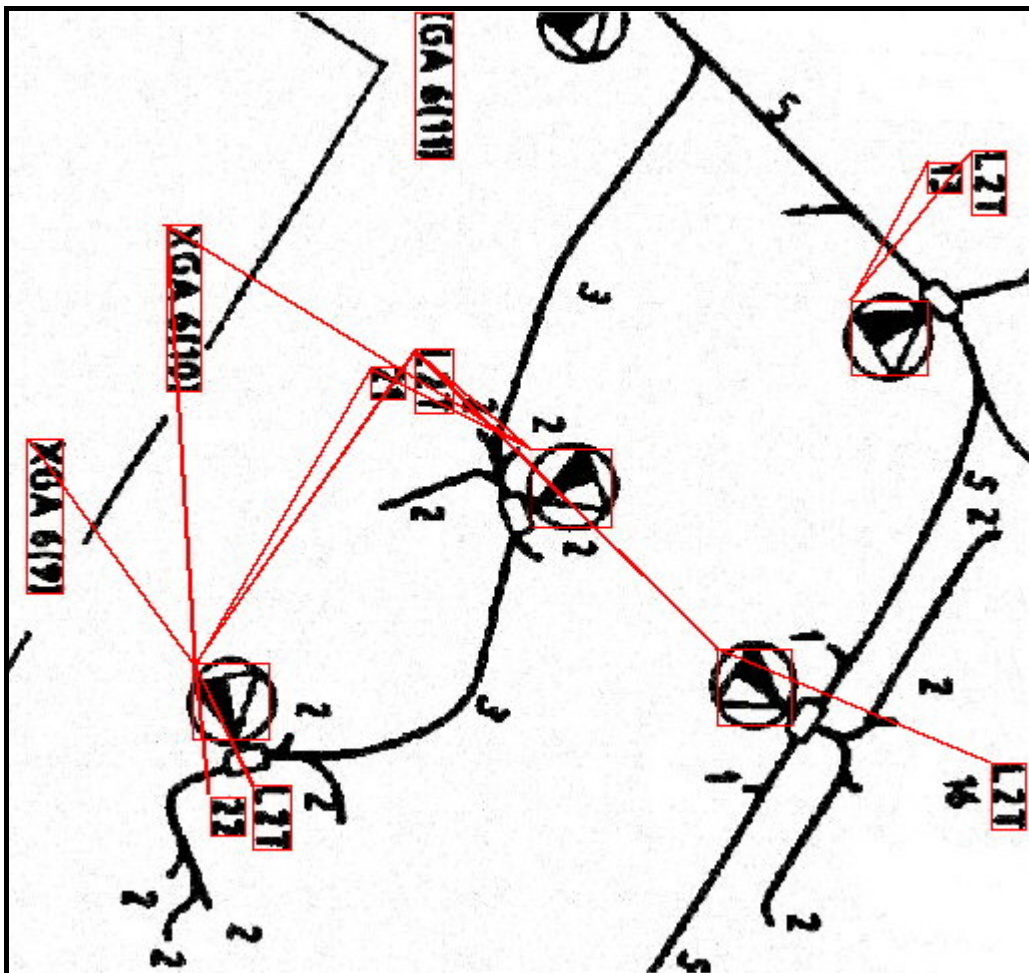


Figure C.7 : Les hypothèses de PC émises

Après cette émission d'hypothèse, le module de validation de cohérence est exécuté. Il est chargé de la vérification de la syntaxe des chaînes de caractères associées aux PC. Lorsque ce travail a été effectué, les PC sont stockés dans la base de données avec un attribut de cohérence et il est alors possible de les visualiser en fonction du scénario utilisé. Dans l'exemple de cette annexe, deux PC complets sont reconstruits. Ils sont montrés sur la figure C.8 avec le scénario exploitant la TFM. Les résultats sont identiques pour les deux autres *scenarii*.

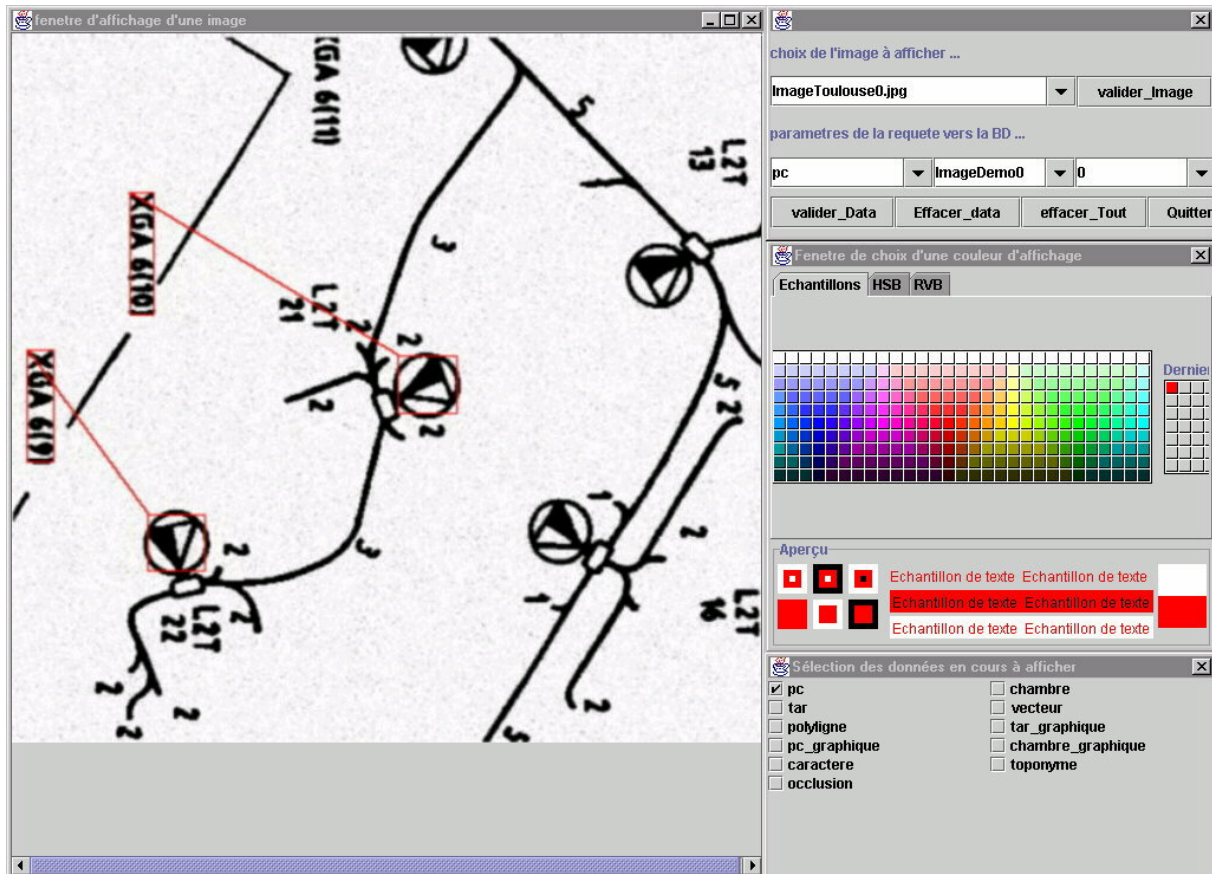


Figure C.8 : Résultat de la requête en utilisant l'OCR basé sur les invariants de Fourier Mellin

Il peut être constaté qu'en toute rigueur, ces résultats ne sont pas corrects. En effet, pour les deux PC reconstruits, les chaînes de caractères sont erronées. Dans un cas, le numéro « 9 » a été confondu avec un « 6 » (ce qui est logique au regard de la méthodologie utilisée). Pour ce problème, nous travaillons actuellement à l'intégration de l'estimation de l'orientation des formes pour la reconstruction des toponymes. La fusion de l'information sur l'orientation du toponyme avec l'orientation des formes devrait permettre de lever de telles ambiguïtés. Dans le second cas, un « R » a été reconnu à la place d'un « 10 » dont les deux caractères sont collés. Cela justifie nos travaux actuels concernant le module de reconnaissance de formes connectées. En effet, un affinement du module de vérification de la cohérence des toponymes peut ici être envisagé en contraignant le système à avoir un chiffre dans les parenthèses dans le cas des identifiants de PC. Lorsqu'il reconnaît un « R » dans une telle chaîne, le

Le système devrait prochainement être en mesure de lancer le « mode filtrage » à partir de la zone d'intérêt correspondante.

Sur la figure C.9, nous résumons les échanges de messages entre agents et les différentes consultations de bases effectués au cours de l'interprétation pour quelques uns des agents concernés.

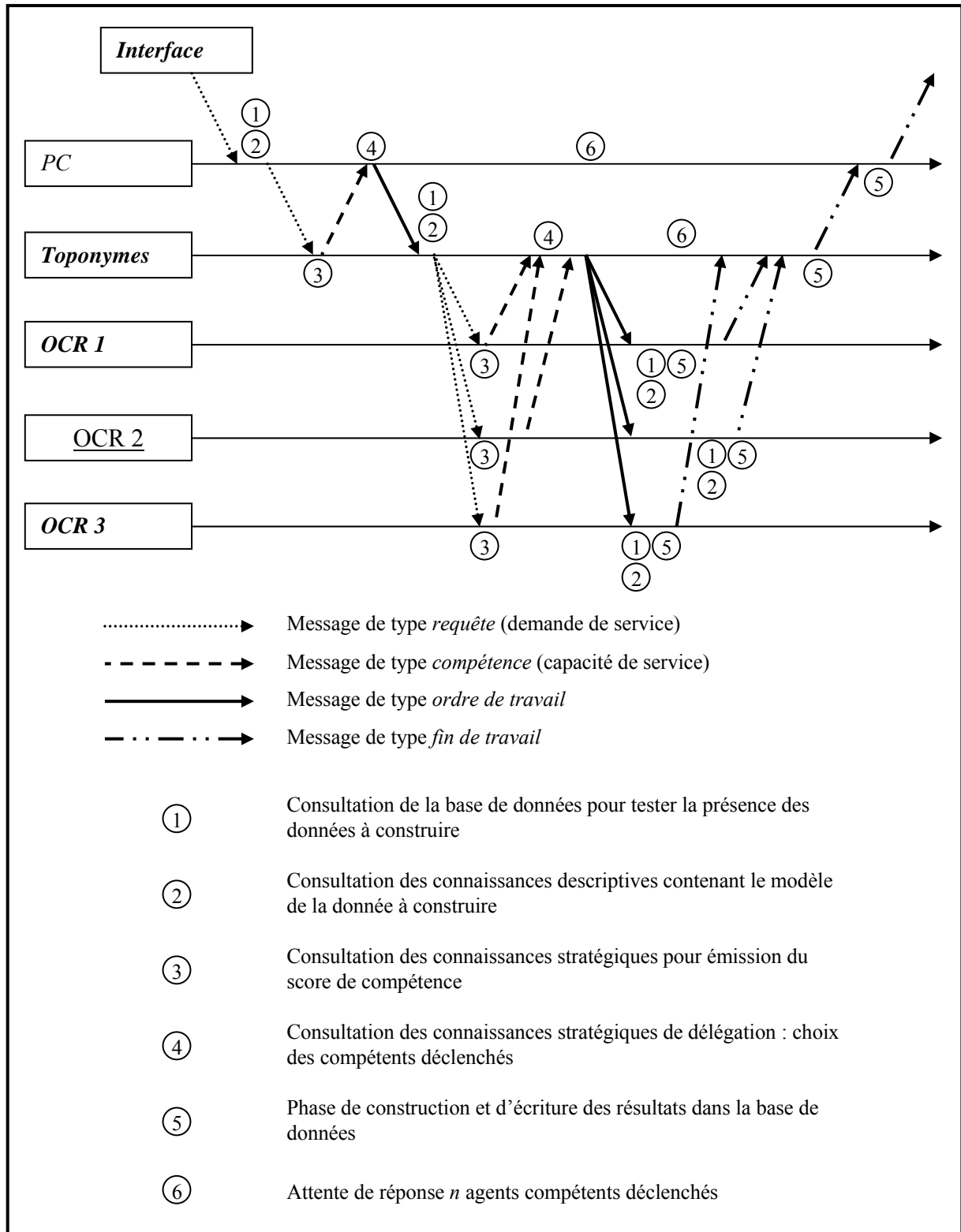
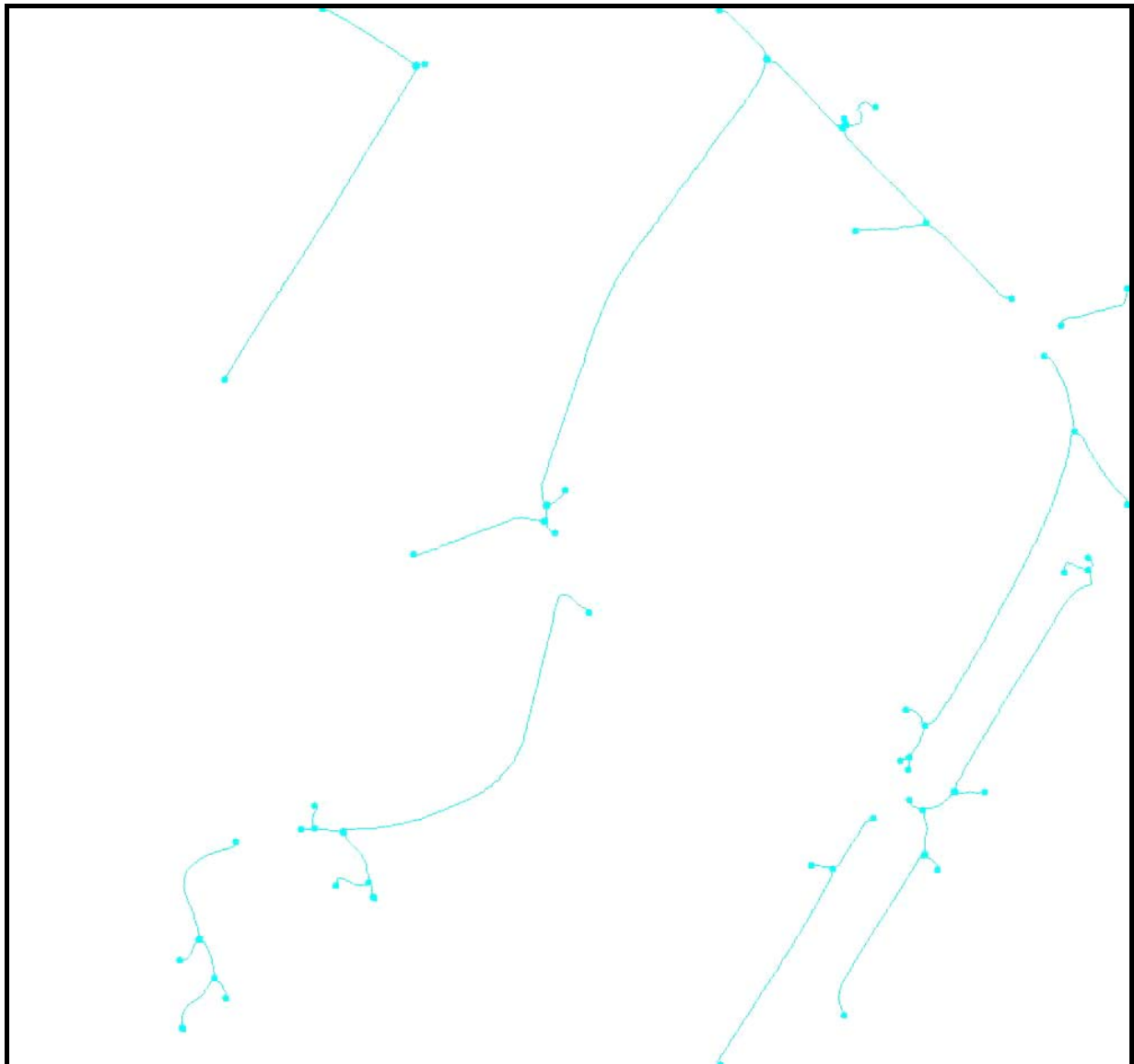


Figure C.9 : Chronogramme des envois de messages et des consultations des bases

En conclusion, si les résultats sont encore loin d'être parfaits, l'ensemble des outils et de l'approche développée semblent en mesure d'apporter de nombreuses solutions aux problèmes actuels de notre équipe..

Notons enfin que la reconnaissance des PC-graphiques (et des chambres\_graphiques) nous permet d'effectuer une vectorisation « propre » de l'ensemble du réseau en utilisant les outils de la littérature qui sont détaillés dans [Adam 2000a]. Les résultats obtenus sont présentés sur la Figure C.9. On peut y remarquer là encore le problème des caractères collés. Comme nous l'avons précisé dans la construction, l'une de nos préoccupations actuelles concerne la détection de zones d'intérêt sur une telle image pour lancer le mode filtrage afin de détecter les caractères collés.



*Figure C.10 : Illustration de la vectorisation du réseau après reconnaissances des symboles*

---

# Table des Illustrations

## Les Figures

Figure 1.1 : Quelques exemples de documents techniques .....	2
Figure 2.1 : Illustration des contraintes imposées au système de reconnaissance.....	11
Figure 2.2 : Un extrait du CCTP 1592, décrivant l'ensemble de la symbologie France Telecom .....	12
Figure 2.3 : Un Modèle générique de la reconnaissance d'une forme isolée utilisant un modèle unique.....	14
Figure 2.4 : Illustration d'une reconstruction d'ordre 2 à partir des moments de l'image .....	20
Figure 2.5 : Synoptique de l'extraction du modèle de Chen [Chen 1999].....	28
Figure 2.6 : Illustration d'histogrammes de projection : à gauche, [Trier 96], à droite [Lee 1991].....	30
Figure 2.7 : Construction du descripteur circulaire de la forme selon Taza [Taza 1989].....	35
Figure 2.8 : Construction des listes caractéristiques selon [Lefrère 1993].....	35
Figure 2.9 : Illustration des profils ( $x_L$ est le profil gauche, $x_R$ le profil droit ([Trier 1996]).....	37
Figure 2.10 : Principes de la méthode de Kimura [Kimura 1991][Trier 1996].....	38
Figure 2.11 : Principes de construction des descripteurs de Fourier.....	39
Figure 2.12 : Illustration intuitive des ellipses dans les descripteurs elliptiques de Fourier .....	40
Figure 2.13 : Illustration du squelette (à droite) d'une image de caractères.....	42
Figure 2.14 : Les parties réelles (à gauche) et imaginaires (à droite) de $h_{p,q}$ avec $p=2$ , $q=3$ , et $\sigma_0=2$ .....	52
Figure 2.15 : Schéma de principe du réarrangement des filtres de FM.....	56
Figure 2.16 : Schéma de principe du réarrangement de l'image .....	56
Figure 2.17 : Deux illustrations de la non convexité du critère $E_{f,g}(\alpha\beta)$ défini en (95).....	57
Figure 2.18 : Un extrait d'une base synthétique de caractères multi-orientés.....	59
Figure 2.19 : Superposition des invariants issus des différentes orientations et des différentes tailles.....	59
Figure 2.20 : Illustration de la complétude de la TFM : reconstruction par la TFM discrète inverse .....	60
Figure 2.21 : Illustration de l'estimation de l'orientation sur le cercle trigonométrique.....	61
Figure 2.22 : Une image de caractères pour l'illustration du mode filtrage .....	62
Figure 2.23 : Illustration des résultats obtenus par le « mode filtrage ».....	62
Figure 2.24 : Courbe taux de confusion / taux de lecture en faisant varier le nombre de voisins .....	73
Figure 2.25 : Illustration des résultats du filtrage sur des formes de synthèse .....	75
Figure 2.26 : Exemples de caractères collés sur les plans France Telecom .....	76
Figure 2.27 : Illustration du traitement des formes collées .....	76
Figure 2.28 : Evolution du score du meilleur individu lors du déroulement de l'algorithme génétique .....	80
Figure 3.1 : Le cycle de vie d'un document suivant [Lorette 1992] [Labiche 1998] .....	88
Figure 3.2 : Le réseau sémantique de Den Hartog .....	94
Figure 3.3 : Les relations de sous partie entre les schémas du réseau .....	96
Figure 3.4 : Les relations d'héritage dans ANON.....	96
Figure 3.5 : La grammaire de définition des symboles dans Ah Soon [Ah Soon 1998] .....	100
Figure 3.6 : Un modèle en couches des données « métier ».....	101
Figure 3.7 : Résolution des incohérences dans Syradoc : le cycle perceptif.....	103
Figure 3.8 : Stratégie, tâches et spécialistes dans CELESSTIN .....	105
Figure 3.9 : Hiérarchie des classes d'objets dans [Ramel 1998].....	106
Figure 3.10 : Illustration des graphes représentant les connaissances du domaine dans [Ramel 1998].....	107
Figure 3.11 : Les connaissances stratégiques dans [Ramel 1998] : un cycle perceptif .....	108

Figure 3.12 : illustration des formalismes proposés par SMECI et utilisés dans MESSIE II.....	119
Figure 3.13 : Vision globale du système BORG [Clouard 1999].....	120
Figure 3.14 : Les modèles de requêtes dans Borg.....	121
Figure 3.15 : Scission spécifique / générique du modèle de connaissances descriptives.....	126
Figure 3.16 : L'opérationnalisation des connaissances descriptives dans NATALI V1.....	128
Figure 3.17 : Les agents implémentés dans NATALI v1.....	129
Figure 3.18 : Les modules d'un agent dans NATALI v1.....	133
Figure 3.19 : L'architecture logicielle de NATALI v1.....	133
Figure 3.20 : Une vision du système d'interprétation du point de vue de l'ingénierie des connaissances.....	137
Figure 3.21 : Hiérarchie des classes dans la bibliothèque de gestion de graphes.....	139
Figure 3.22 : La partie « définition des objets » pour les connaissances du domaine.....	140
Figure 3.23 : Les relations existant entre objets élémentaires sur les plans de réseau.....	142
Figure 3.24 : L'interface d'acquisition du réseau pour la représentation des connaissances du domaine.....	143
Figure 3.25 : Illustration de la génération automatique de la structure du SMA.....	144
Figure 3.26 : Les stratégies possibles de déclenchement d'opérateurs.....	147
Figure 3.27 : Les stratégies possibles de gestion des résultats.....	148
Figure 3.28 : L'interface d'acquisition des connaissances descriptives graphiques appliquées aux caractères..	151
Figure 3.29 : Un exemple jouet, avant et après correction et les performances correspondantes.....	153

## Les Tableaux

Tableau 2.1 : Les données utilisées dans le comparatif de Belkasim [Belkasim 1991].....	32
Tableau 2.2 : Synthèse des modèles de la littérature.....	44
Tableau 2.3 : Nombre de prototypes par classe dans la base de synthèse.....	67
Tableau 2.4 : Nombre de prototypes par classe dans la base France Telecom.....	68
Tableau 2.5 : Nombre de prototypes par classe dans la base EDF.....	68
Tableau 2.6 : Nombre de prototypes par classe dans la base d'apprentissage MNIST.....	69
Tableau 2.7 : Nombre de prototypes par classe dans la base de test MNIST.....	69
Tableau 2.8 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base de synthèse.....	69
Tableau 2.9 : Comparaison entre les modèles sur la base de synthèse avec le LVQ.....	70
Tableau 2.10 : Les confusions du test du 1-PPV avec le modèle de FM.....	70
Tableau 2.11 : Comparaison entre les modèles sur la base de synthèse avec le 1-PPV.....	70
Tableau 2.12 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base FT.....	71
Tableau 2.13 : Comparaison entre les modèles sur la base France Telecom avec le LVQ.....	71
Tableau 2.14 : Taux de reconnaissance par classe obtenus par FM et 1 ppv sur la base FT.....	72
Tableau 2.15 : Matrice de confusion du test du 1-PPV avec le modèle de FM.....	72
Tableau 2.16 : Comparaison entre les modèles sur la base France Telecom avec 1 ppv.....	72
Tableau 2.17 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur la base EDF.....	74
Tableau 2.18 : Taux de reconnaissance par classe obtenus par FM et le LVQ sur NIST.....	74
Tableau 2.19 : Taux de reconnaissance par classe obtenus par Zernike et le LVQ sur NIST.....	74
Tableau 2.20 : Taux de reconnaissance par classe obtenus par les sondes circulaires et le LVQ sur NIST.....	74
Tableau 2.21 : Comparaison entre les modèles sur la base NIST avec le LVQ.....	74
Tableau 2.22 : L'estimation du facteur d'échelle et de l'orientation.....	77
Tableau 2.23 : L'estimation de l'orientation : comparaison entre Fourier Mellin et Zernike.....	78
Tableau 3.1 : Le modèle d'une parcelle.....	102
Tableau 3.2 : Table de résolution des incohérences.....	103
Tableau 3.3 : Les connaissances dans les systèmes d'interprétation de documents techniques.....	108