

Exploitation de connaissances « domaine » pour l'analyse et l'interprétation d'images Application à l'indexation sémantique

Nicolas Zlatoff
Stage de DEA, sous la direction de Bruno Tellez
Avril – Juillet 2003

**DEA Documents Multimédia et Systèmes d'Informations Communicants (DISIC),
Ecole Doctorale Informatique et Information pour la Société (IIS)**



Laboratoire d'InfoRmatique en Images et
Systèmes d'information (LIRIS)



Maison de l'Orient et de la Méditerranée
(MOM)

Résumé

Pour être réellement efficaces, les outils d'indexation d'images doivent être automatiques en terme d'analyse et de description du contenu. Le fossé sémantique existant entre les concepts haut-niveau et des descripteurs bas-niveau (couleur, texture), rend inévitable l'introduction dans le processus de connaissances externes. Cette étude s'intéresse à l'exploitation de connaissance « domaine » pour l'interprétation d'images. Plus précisément, nous nous intéressons à l'indexation d'images de stèles thessaliennes, fournies par la Maison de l'Orient et de la Méditerranée (MOM). Cette étude présente une méthodologie qui permet d'identifier des objets à partir de régions segmentées. Ces objets sont ensuite interprétés grâce aux connaissances « domaine ». Pour obtenir ce résultat, nous insisterons sur la nécessité de séparer connaissances et raisonnement. Ainsi, les connaissances (concepts et relations du domaine) sont stockées dans une ontologie à base de frames, en tant que faits. Enfin, d'autres connaissances sont introduites sous forme de règles sur lesquelles vont s'appliquer les raisonnements.

Mots Clef

Interprétation d'image, connaissances, ontologie, inférences, indexation d'images.

Cette étude a fait l'objet d'une soumission de publication à la conférence RFIA 2004 (Reconnaissance de Formes et Intelligence Artificielle).

Abstract

In order to be truly efficient, indexing image tools have to be automatic for content analysis and description. The semantic gap between high-level concepts and low-level descriptors, such as colour or texture, leads to the necessary introduction of knowledge from an external source. This study focuses on "domain" knowledge guiding an image understanding process. More precisely, we deal with Thessalian graves images from the "Maison de l'Orient et de la Méditerranée" (MOM). This study presents a methodology to identify objects from homogeneous segments. These objects will be next semantically characterized thanks to "domain" knowledge. In such a way, we emphasize the necessity of a separation between knowledge and reasoning. Thus, domain knowledge will be stored in a frame-based ontology, as facts. In the same way, additional knowledge has been introduced as rules. Reasoning schemes are then applying on these rules.

Keywords

Image understanding, knowledge, ontology, inferences, image indexing.

Table des matières

I. INTRODUCTION	3
A. L'INDEXATION D'IMAGES	3
B. LA MAISON DE L'ORIENT ET DE LA MEDITERRANEE (MOM)	4
II. ETAT DE L'ART SUR LA MODELISATION ET L'EXPLOITATION DE CONNAISSANCES « DOMAINE »	6
A. SEPARER CONNAISSANCES ET RAISONNEMENT	6
B. LES SYSTEMES A BASE DE REGLES	7
1. PRINCIPES	7
2. APPLICATIONS	8
C. LES RESEAUX SEMANTIQUES	8
1. PRINCIPES	8
2. APPLICATIONS	9
D. LES GRAPHES CONCEPTUELS	9
1. PRINCIPES	9
2. APPLICATIONS	10
E. LES SYSTEMES A BASE DE FRAMES	10
1. PRINCIPES	10
2. APPLICATIONS	11
F. LES TRAVAUX DU WEB SEMANTIQUE	11
1. LES LANGAGES RDF, RDFS PUIS DAML+OIL	12
2. ONTOSEEK	13
G. CONCLUSION SUR L'ETAT DE L'ART	13
III. METHODOLOGIE : COLLABORER ENTRE HAUT-NIVEAU ET BAS-NIVEAU	14
A. PRESENTATION DE LA METHODOLOGIE	14
B. LES TRAITEMENTS BAS-NIVEAU	15
1. DESCRIPTEURS DE TEXTURE ET CLUSTERING	15
2. DISCUSSION ET VALIDATION	16
C. LA MODELISATION ET LE STOCKAGE DES CONNAISSANCES	18
1. MODELISATION	18
2. STOCKAGE DANS UN SYSTEME A BASE DE FRAMES	19
D. INFERER DES CONNAISSANCES SEMANTIQUES	21
E. IMPLANTATION	23
1. MODELISATION ET INTRODUCTION DE CONNAISSANCES	23
2. RAISONNEMENT ET INFERENCE	24
IV. CONCLUSION ET PERSPECTIVES	26
V. REMERCIEMENTS	27
VI. BIBLIOGRAPHIE	27

I. Introduction

A. L'indexation d'images

Durant ces dix dernières années, l'informatique a connu une augmentation considérable des capacités de stockage, couplée à une interconnexion des réseaux sans cesse croissante. Par conséquent, un nombre gigantesque de documents numériques et plus particulièrement d'images sont maintenant disponibles en libre accès ou dans des collections privées.

Dès lors, pour pouvoir correctement exploiter ces banques d'images, il est indispensable de disposer d'outils qui permettront d'identifier les images répondant aux requêtes des utilisateurs. Cette identification se fait soit par similarité soit par identification du contenu de l'image. Cette dernière procédure dite d'indexation est une tâche extrêmement fastidieuse lorsqu'elle est effectuée manuellement. Dès lors, il semble naturel de chercher à automatiser cette tâche lorsque le volume d'images est conséquent.

Pour l'instant, de nombreux outils dits « *content-based* » existent¹ (QBIC [Niblack93], VisualSEEK [Smith96], pour ne citer qu'eux). Ils permettent une indexation automatique d'images, sur des critères bas-niveau comme la couleur, la texture, voire la forme de l'objet principal de l'image. Néanmoins, cette indexation n'est en général pas pertinente pour l'utilisateur qui, dans la plupart des cas, cherche une image d'après ce qu'elle représente (sa sémantique) : un objet, une personne, une scène particulière.

Chercher à étendre les principes des outils « *content-based* » pour essayer de déduire une sémantique à partir de primitives bas-niveau constitue une méthode dite ascendante, qui trouve rapidement ses limites. En effet, il semble impossible d'associer de manière unique un concept haut-niveau à une série de descripteurs bas-niveau, justement parce qu'un concept est une abstraction visant à regrouper différentes formes d'apparition.

Cet absence de lien direct entre haut-niveau et bas-niveau est connu sous le nom de **fossé sémantique**.

Dès lors, pour tenter de combler ce fossé, il apparaît nécessaire d'introduire des connaissances extérieures lors du processus d'indexation. On distinguera à ce stade deux types de connaissances : d'une part les connaissances « vision », relatives aux mécanismes de perception en tant que tels et d'autre part les connaissances « domaine », relatives au domaine dénoté par l'image, et portant sur des objets sémantiques.

L'étude présentée dans ce rapport s'intègre dans des travaux de recherche visant à l'élaboration d'une méthodologie d'indexation d'image en exploitant des connaissances « domaine ». Etant donné la complexité du problème, il s'agira surtout de montrer la faisabilité d'une telle approche et de proposer une méthodologie.

On distinguera dans le processus, d'une part l'analyse d'images qui, à partir d'une image segmentée, consiste à exploiter des primitives bas-niveau de l'image (segments) pour reconnaître des objets sémantiques, sans qu'il soit nécessaire de les nommer à ce stade et d'autre part, l'interprétation d'images, qui reconnaît effectivement les objets générés par l'analyse d'image.

Le « domaine » en question sera celui de l'archéologie, plus précisément l'étude des stèles thessaliennes. En effet, ce travail s'inscrit dans le cadre d'une collaboration avec la Maison de l'Orient et de la Méditerranée (MOM), qui dispose d'une série d'images que les archéologues souhaiteraient indexer sémantiquement de manière automatique.

¹ Le terme « *content-based* » est mal adapté, puisqu'il renvoie à des notions bas-niveau comme la couleur et non pas à des contenus sémantiques.

B. La Maison de l'Orient et de la Méditerranée (MOM)

La Maison de l'Orient et de la Méditerranée (MOM) dispose actuellement d'une banque non indexée de plusieurs dizaines de milliers d'images représentant des stèles thessaliennes. Ces images sont issues de la numérisation de photographies argentiques prises par l'Ecole Française d'Athènes il y a une dizaine d'années. Devant la taille imposante de cette base d'images et dans le but d'accéder correctement aux informations de cette base, la MOM souhaiterait disposer d'outils permettant d'automatiser l'indexation.

Or, les conditions de prises de vues sont globalement de trop mauvaise qualité pour utiliser des méthodes traditionnelles de traitement d'images, et posent donc rapidement de réels problèmes en terme d'automatisation des tâches.

La Figure 1 présente quelques exemples d'images de stèles.

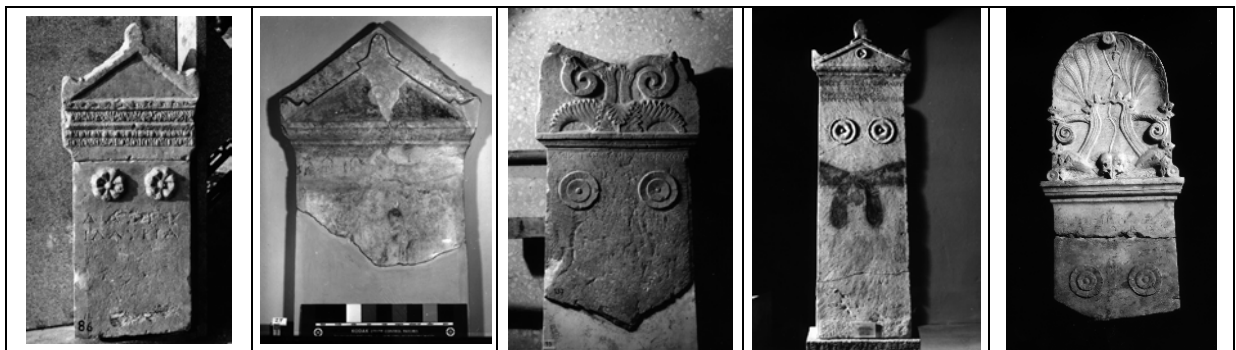


Figure 1- Exemple d'images de stèles

Une stèle funéraire est un bloc de pierre, peint et / ou sculpté, enfoncé dans le sol afin de marquer l'emplacement d'une sépulture. En première approche, une stèle Thessalienne comporte deux types d'informations (voir à ce titre le Tableau 1): celles liées à la forme générale, impliquant des éléments particuliers (corps de stèle, couronnement, geison) et celles liées à des travaux de finition comme les ornements, les inscriptions, etc...

La Figure 2 présente quelques exemples d'objets caractéristiques d'une stèle.

Forme générale (éléments blocs)	Finition
Couronnement, geison, corps de stèle, socle	Ornement (rosette, paire de rosettes, tainia), inscriptions

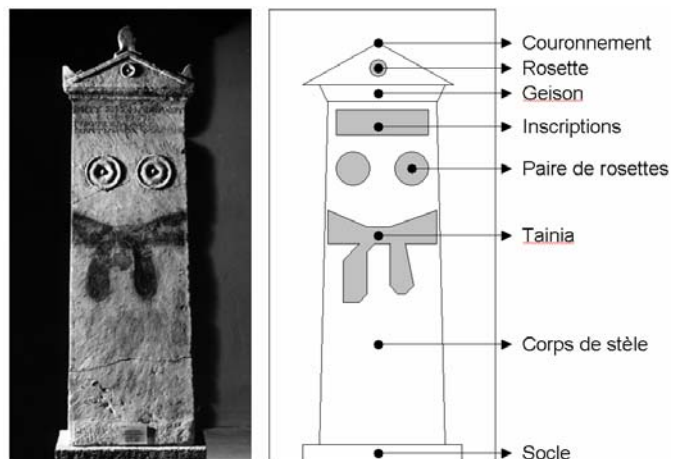


Tableau 1- Différents types de composants d'une stèle

Figure 2- Exemple de composants d'une stèle

La Figure 3 présente une illustration des deux procédés d'analyse et d'interprétation d'image, dans le domaine que nous avons choisi : le bas-niveau est constitué de plusieurs segments, caractérisés par des descripteurs bas-niveau. Le haut-niveau comprend un modèle générique de connaissances « domaine », ainsi qu'un modèle instancié de connaissances, valable pour l'image en cours de traitement.

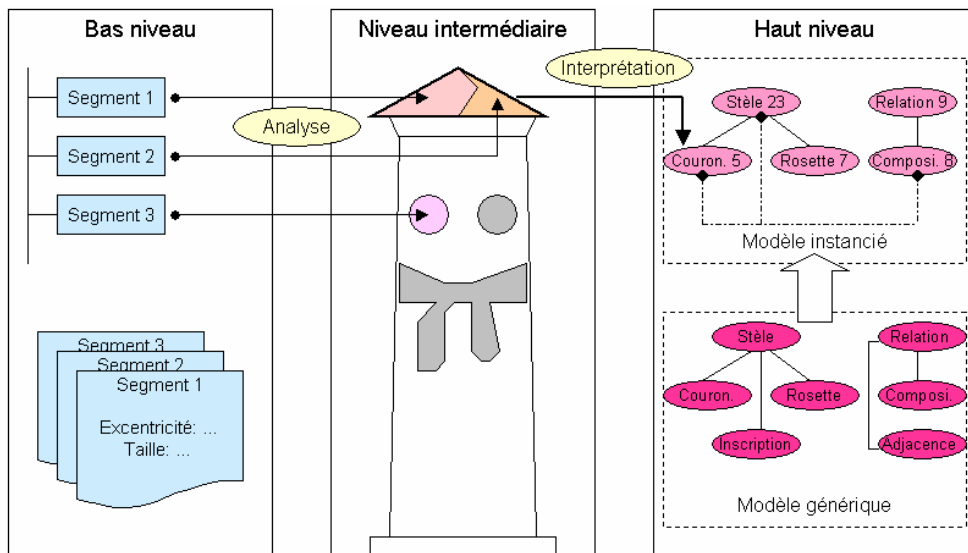


Figure 3- Illustration des trois niveaux de l'image : du segment à l'objet sémantique

Dans une première partie de ce rapport, nous présentons un état de l'art sur la modélisation et l'exploitation de connaissances « domaine », appliquées au problématiques d'analyse et d'interprétation d'images. La partie suivante exposera les différents travaux menés dans le cadre du stage, concernant les descripteurs bas-niveau nécessaires à la segmentation, mais aussi toute la phase de formalisation et d'exploitation des connaissances « domaine » pour interpréter des objets d'intérêt, sélectionnés manuellement dans les images. En outre, nous consacrerons une sous-partie aux implantations effectuées.

Enfin, nous commenterons ce travail et présenterons les perspectives futures dans une dernière partie.

II. Etat de l'art sur la modélisation et l'exploitation de connaissances « domaine »

De nombreux travaux tentent d'exploiter des connaissances « domaine » dans le but d'assister ou de piloter un processus d'analyse et d'interprétation d'images. Par exemple, [Huertas96], dans le cadre du projet RADIUS, propose un système de détection des changements dans des bâtiments, suivant une approche descendante. L'image à traiter est rattachée à un modèle de site, puis, les différents objets présents dans le modèle sont effectivement recherchés dans l'image, en comparant les caractéristiques extraites des caractéristiques prévues par le modèle. Un écart notable de ces valeurs est interprété comme un possible changement, qui est alors confirmé ou infirmé en étudiant les différentes situations possibles : objets occultés, ombres projetées par exemple, ou bien changement effectif.

Une approche mixte, décrite dans [Mees96] propose un système piloté par un tableau noir, d'aide à l'interprétation de photos. Un processus spécialiste génère ainsi des primitives bas-niveau à partir de l'image, qui servent à définir des solutions partielles. Un processus descendant est alors appelé pour vérifier la cohérence des objets d'intérêt issus de la phase précédente. Enfin, le processus est relancé à un niveau de détail plus fin.

Le point commun entre ces systèmes, ainsi qu'entre la plupart des systèmes d'analyse ou d'interprétation d'images, est qu'ils fonctionnent de manière procédurale, en liant représentation des connaissances et raisonnement. Dans ce cas, la manière de résoudre le problème est contenue dans le code même du système : un enchaînement d'appels à des procédures qui manipulent des structures de données figées. Ces systèmes ne sont efficaces que si la manière de résoudre le problème est connue a priori. Or, la méthodologie proposée ne cherche pas à fixer de cadre procédural. Au contraire, elle vise à construire un système ouvert qui permette à des experts du domaine de modéliser leurs connaissances, puis de définir les raisonnements qui leur seront appliqués. L'objectif visé est de fournir un mécanisme de révision qui permette d'aboutir à une conclusion mais qui, si elle n'est pas satisfaisante, pourra être remise en cause. Ces raisonnements concernent aussi bien les connaissances du domaine que les connaissances spécifiques à la vision.

A. Séparer connaissances et raisonnement

Pour pallier aux limitations de la programmation procédurale, l'intelligence artificielle propose une programmation déclarative, qui sépare représentation des connaissances et raisonnement. Ce dernier peut alors, à partir des connaissances déjà stockées, inférer, déduire de nouvelles connaissances.

Les connaissances stockées constituent une **ontologie**. En effet, d'après [Gruber93], une ontologie est une « *explicite spécification d'une conceptualisation admise* ». Une ontologie regroupe donc des connaissances (c'est-à-dire les concepts et les relations dans le monde étudié) sous un formalisme donné.

[Noy01a] précise qu'il n'existe pas une unique ontologie pour un problème donné. Néanmoins, [Uschold95] donne une méthodologie de construction en quatre étapes. La première, fondamentale, consiste à définir l'usage futur qui sera fait de l'ontologie. Dans l'architecture proposée, l'ontologie devra permettre d'exploiter des connaissances « domaine » pour analyser et interpréter des segments d'images. On rejoint donc une problématique de classification : étant donnés différents segments, comment les assembler pour former des objets présentant une cohérence sémantique ? Puis, étant donnés des objets sémantiques, dans quelle catégorie du modèle les placer ?

La deuxième étape consiste ensuite à construire l'ontologie en tant que telle : identifier les concepts et relations du discours, et transcrire ces notions dans un formalisme.

Enfin, l'ontologie produite doit être évaluée et documentée.

Concernant l'étape de construction, plusieurs voies sont possibles. Par exemple, [Uschold95] commence par identifier les concepts clés et les relations entre ces concepts, fournissant ainsi une première modélisation. Cette phase est ensuite suivie par la définition de chacun des concepts et par la

sélection de tous les termes désignant le concept. [Fox94] propose une voie plus orientée sur l'étude des variables et constantes qui représentent dans un premier temps les objets de l'ontologie, aidant ainsi à leur compréhension.

Quoi qu'il en soit, cette phase de construction de l'ontologie doit permettre d'identifier les concepts et relations de l'ontologie, de leur donner une définition et de s'accorder sur un vocabulaire adapté.

Une fois l'ontologie définie à ce niveau, il s'agit de la stocker sous un formalisme. Nous présentons maintenant les principaux formalismes disponibles en nous appuyant sur les états de l'art de [Roussey01] et [Drews93]. En outre, pour chaque formalisme, nous présentons les méthodes de raisonnement associées. En effet, il faut bien voir que, même si connaissances et raisonnements sont séparés, chaque représentation de connaissances dispose d'outils de raisonnements privilégiés.

B. Les systèmes à base de règles

Introduits dans les années 70, ils représentent une évolution naturelle des systèmes basés sur la logique du premier ordre, premiers systèmes historiquement à avoir séparé connaissances et raisonnements.

1. Principes

La base de connaissance est constituée d'une base de faits, comprenant plusieurs formules logiques. Celles-ci sont formées à partir de prédicats, de constantes et de variables. Les faits constituent un contexte et représentent l'état actuel des connaissances du système.

Exemples de faits :

Rosette(objet48)	« objet48 est une rosette »
Corps(objet2)	« objet2 est un corps »
Rond(objet45)	« objet45 est rond »

De plus, la base de connaissances comporte un ensemble de règles du type

SI <condition> ALORS <action>.

Le terme <condition> est un prédicat logique. Si un des faits de la base de connaissances vérifie ce prédicat (ie : si la règle peut s'unifier avec un des faits), la règle est activée, et la partie <action> pourra être déclenchée. Cette partie consiste soit en une modification de la base de faits, soit éventuellement en l'appel d'une procédure externe.

Exemple de règle :

Règle 1 : Si rosette(x), alors rond(x)
Signifie : « si un objet x est une rosette, alors x est rond »

La partie raisonnement est constituée par le moteur d'inférences, qui contrôle l'application des règles. Il détecte d'abord quelles règles voient leur <condition> être vraie pour certains faits : le moteur vérifie si <condition> peut être unifiée avec certains faits de la base de connaissance. On dit dans ce cas, qu'une règle est *activée*. Dans notre exemple, la condition `rosette(x)` serait unifiée au fait `Rosette(objet48)` et la règle 1 serait activée.

Si plusieurs règles ont été activées, le moteur d'inférences choisit laquelle appliquer. Ce choix peut être déterminé en fonction des nouveaux faits appris par exemple.

Un moteur d'inférences fonctionne selon deux modes de raisonnement : le **chaînage avant** d'une part consiste à exploiter le mécanisme précédent et à enrichir la base de faits par de nouveaux faits. Le processus s'arrête lorsque aucune règle ne peut plus s'unifier à aucun fait. On dit que le chaînage avant est dirigé par les données.

Le **chaînage arrière** quant à lui, est dirigé par un but : l'utilisateur demande au système de prouver un but, c'est-à-dire un fait. Dans ce cas, le mécanisme de recherche de règles applicables est modifié et fonctionne en cherchant à unifier la partie <action> des règles avec le but à prouver. Le processus s'arrête lorsqu'il n'y a plus aucun but à prouver.

2. Applications

Prolog, mais aussi Jess sont des exemples bien connus de systèmes à base de règles. Les applications ne manquent pas, notamment dans le domaine médical. Ainsi MYCIN [Shortliffe76], dédié au diagnostic médical, est considéré comme un des systèmes pionniers.

[Li96] utilise un système à base de règles pour déduire des relations topologiques et directionnelles fines entre segments d'images, à partir de relations principales.

Les systèmes à base de règles sont efficaces pour déployer des raisonnements de diagnostic, basés sur la causalité, à partir d'objets simples. Néanmoins, le problème fondamental de ce type de système réside dans le fait qu'ils ne permettent pas de structurer réellement la connaissance : la base de connaissance est composée d'une suite de faits et de règles non organisés entre eux. Ainsi, la connaissance relative à un objet est disséminée dans la base de connaissance. En outre, toute relation entre concepts (comme la spécialisation par exemple) doit être formalisée explicitement.

C. Les réseaux sémantiques

Les réseaux sémantiques constituent le premier modèle à héritage utilisé en représentation des connaissances, issus de travaux de psychologie cognitive sur l'organisation de la mémoire. Les graphes conceptuels [SOWA84] ou encore le langage KL-ONE sont des exemples bien connus de réseaux sémantiques.

1. Principes

Les réseaux sémantiques sont classiquement définis par [Quillian68] en tant que graphe dont les nœuds représentent les concepts du domaine modélisé. Les arcs entre ces nœuds sont étiquetés et orientés et représentent les relations binaires (c'est-à-dire impliquant deux concepts) qui existent entre les concepts.

Plus précisément, un réseau sémantique est un multigraphe, dans le sens où deux nœuds peuvent être reliés par deux arcs différents.

Dans le cas où une relation n'est pas binaire, elle est modélisée par un concept en tant que tel.

Deux relations binaires sont particulièrement importantes : la relation de spécialisation (entre un concept générique et un concept plus spécialisé) et la relation d'instanciation (entre un concept générique et un concept individuel).

La Figure 4 présente un exemple de réseau sémantique pour des constituants de la stèle.

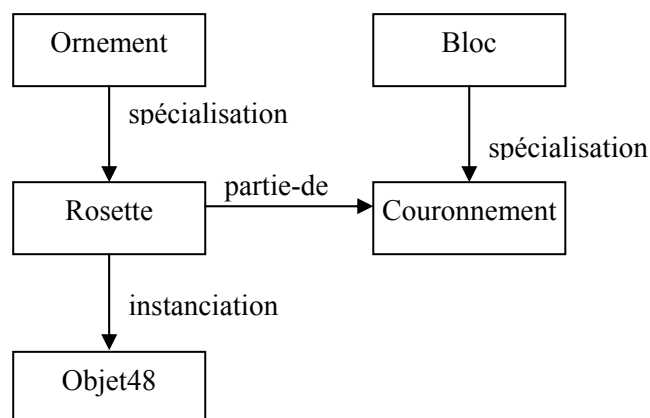


Figure 4- Exemple de réseau sémantique

Raisonnement par **filtrage** avec un réseau sémantique consiste à faire une recherche de sous-graphe dans la base de connaissances : la question est modélisée par un graphe, dans lequel les données inconnues sont représentées par des variables. Le système cherche alors à appairer la question avec une partie de la base de connaissances. La réponse est obtenue lorsque l'appariement substitue aux variables une valeur, issue de la base de connaissances.

Le raisonnement par filtrage peut en outre s'appuyer sur l'opération **d'inférence par héritage**, qui consiste à transférer la connaissance relative à un concept générique à tous les concepts instances du concept générique.

2. Applications

[Kunz97] propose un système d'interprétation d'image satellites, en comparant celles-ci avec les images d'une base topographique digitale, représentant le même espace. Le modèle a priori de ce dernier est formalisé au niveau sémantique via des réseaux sémantiques dans le système ERNEST [Kummert93] et l'extraction des objets dans l'image satellite est alors pilotée par la connaissances contenue dans les réseaux. En outre, la comparaison entre les connaissances a priori et celles extraites de l'image est réalisée en exploitant directement les outils liés aux réseaux sémantiques comme le filtrage.

Les réseaux sémantiques permettent une première structuration de la connaissance, en introduisant des relations fondamentales comme la spécialisation, l'instanciation. Néanmoins, certains inconvénients apparaissent assez vite, comme par exemple l'absence de quantificateurs. En outre, la modélisation des relations est un peu hasardeuse puisque les relations binaires sont représentées par des arcs alors que les autres sont modélisées par un concept en tant que tel. Enfin, il est important de noter que la connaissance relative à un objet sémantique est disséminée dans tout le réseau.

D. Les graphes conceptuels

Introduits par [Sowa84], les graphes conceptuels avaient pour but premier la modélisation des connaissances associées au langage.

1. Principes

Un graphe conceptuel est composé de deux sortes de nœuds : les concepts et les relations. C'est donc un graphe bipartite. En outre, les concepts et les relations sont typés.

Plus précisément, un nœud concept est étiqueté par un type et par un marqueur. Ce dernier permet d'identifier un individu relatif au type. Si on ne souhaite pas préciser d'individu en particulier, on dispose du quantificateur universel (*), qui identifie « n'importe quel individu du type ».

Les relations sont, elles, étiquetées par un type.

Les graphes conceptuels organisent en outre les types de concepts dans un treillis, suivant une hiérarchie de spécialisation. Cette hiérarchie définit un ordre partiel, dont le type le plus spécialisé (le plus petit élément au sens de la relation d'ordre partielle) est le type absurde et le type le moins spécialisé (le plus grand élément) est le type universel.

De même, les relations de même arité sont organisées entre elles dans un autre treillis, suivant une hiérarchie de spécialisation. En outre, la signature d'une relation impose des contraintes de types sur les arguments de la relation (les concepts).

La Figure 5 modélise la connaissance « la stèle GHW278 est composée d'une rosette ».

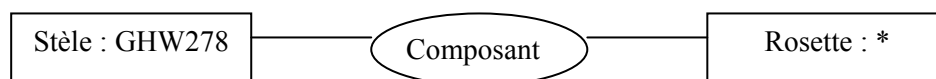


Figure 5- Exemple de graphe conceptuel: « la stèle GHW278 est composée d'une rosette »

Concernant les raisonnements associés aux graphes conceptuels, la relation de spécialisation permet de définir un **opérateur de projection** : il existe un opérateur de projection d'un graphe A dans un graphe B lorsque le graphe B présente des concepts plus spécifiques que les concepts de A. On dit que B est une spécialisation de A ($B < A$).

En outre, un **opérateur Φ** permet d'associer à un graphe conceptuel une formule logique du premier ordre. Cet opérateur vérifie :

$$\text{Si } B < A \text{ alors } \Phi(B) \Rightarrow \Phi(A)$$

2. Applications

[Mulhem00] et considère une image segmentée dans laquelle chaque segment désigne un objet sémantique ou seulement une partie d'un objet sémantique. On dit que chaque segment porte un label. Les auteurs proposent alors un système qui permet de raffiner le taux de confiance accordé à chacun des labels. Un apprentissage est effectué qui permet de modéliser, d'après une base d'exemple, des dispositions spatiales caractéristiques entre les objets sémantiques à l'aide de graphes conceptuels, un type de concept représentant un objet sémantique. Puis, après être passé dans un espace probabilisé via l'opérateur Φ , la théorie de Dempster-Shafer sur les hypothèses composées permet de calculer une nouvelle valeur de confiance (probabilité de réalisation), à partir des données de l'apprentissage, afin de confirmer ou d'infirmer chacun des labels originaux.

[Mechkour95] présente EMIR², un modèle formel de système de recherche d'images qui modélise celles-ci par des graphes conceptuels, suivant différentes vues, éventuellement couplées entre elles : présence d'objets sémantiques, relation spatiales et formes des objets, descripteurs bas-niveau. En outre, les connaissances liées au domaine sont modélisées par les treillis de types de concepts et de relations. Une requête sur la base est elle-même un graphe R. Si la représentation I d'une image est une projection (au sens des graphes conceptuels) de R, alors l'implication $\Phi(I) \Rightarrow \Phi(R)$ est vraie et rend l'image I pertinente pour la recherche.

Les graphes conceptuels poussent donc un peu plus loin encore la structuration de la connaissance des graphes sémantiques, tout d'abord en distinguant nettement concepts et relations mais aussi en introduisant les notions de types et de marqueurs. En outre, les graphes conceptuels organisent la connaissances dans différentes structures, utilisées ensuite « au besoin » : treillis des types de concepts et des relations d'une part, graphes spécifiquement reliés à une connaissance formalisée d'autre part (comme celui de la Figure 5 par exemple).

Historiquement, les graphes conceptuels ont été initiés pour modéliser la connaissance liée au langage. Ceci les rend ainsi naturellement adaptés pour des systèmes d'indexation et de recherche d'informations. Néanmoins, leur formalisme rend assez complexe leur utilisation et n'est pas forcément adapté à des connaissances non orientées langage.

E. Les systèmes à base de frames

Ces systèmes, originellement définis par [Minsky85] regroupe les connaissances en « paquets » (frames) à la différence des réseaux sémantiques. En effet, dans ces derniers, une unité sémantique décrit un concept et ensuite, la définition de ce concept est décrite par d'autres unités sémantiques. Dans un frame au contraire, la connaissance est regroupées en paquets et l'unité sémantique est confondue avec sa définition.

1. Principes

Un **frame** est un prototype qui représente une famille. Il est composé de slots (ou attributs) qui décrivent les propriétés du prototype. Les slots peuvent être à leur tour décrits par des facettes (contraintes sur les slots). Parmi les facettes, on distingue les facettes statiques: précision du type, du domaine, d'une valeur par défaut du slot par exemple ; ainsi que les facettes procédurales, qui servent à associer une procédure à un slot, par exemple pour calculer sa valeur, ou vérifier une contrainte.

Les frames sont organisées dans une hiérarchie suivant une relation de spécialisation. Cette hiérarchie permet de faire hériter à un frame les slots et leurs valeurs depuis le frame parent et évite ainsi la recopie des informations pour chaque frame. En outre, la spécialisation du frame par rapport à ses parents est opérée en lui rajoutant des slots ou en ajoutant des facettes sur les slots hérités.

Une évolution de la description de base des frames a conduit à distinguer deux types de frames : les frames classes qui représentent les catégories d'objets du monde et les frames instances qui représentent les individus.

La Figure 6 présente un exemple de modélisation avec un système à base de frames.

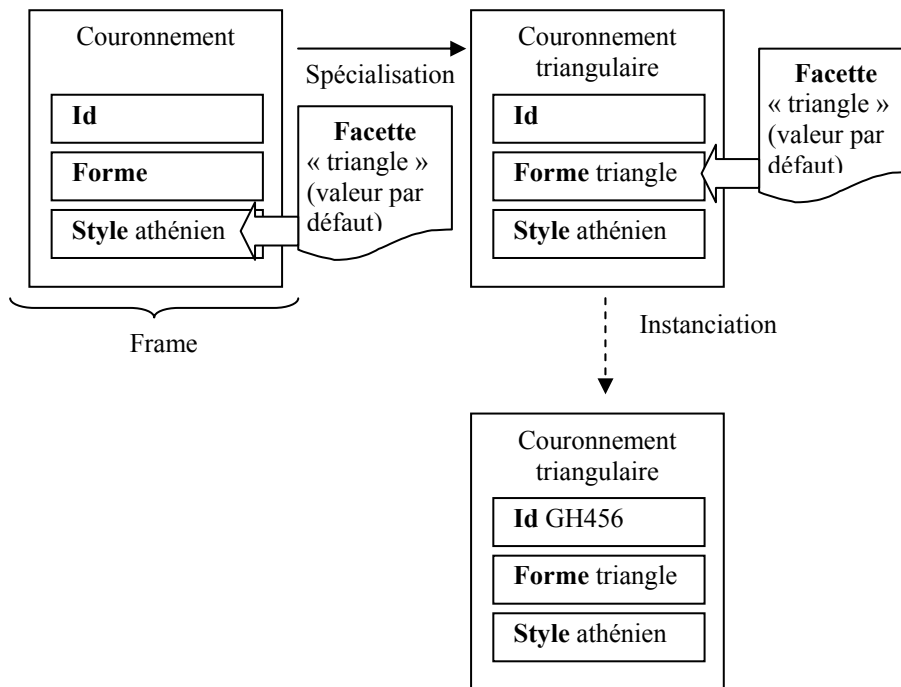


Figure 6- Exemples de frames, organisées suivant les relations de spécialisation et d'instanciation

Fondamentalement, un système à base de frames est dédié au classement, et dispose donc de deux mécanismes adaptés : le filtrage et la classification. Le **filtrage** part d'un ensemble de critères, représenté sous la forme d'un frame filtre, et recherche le frame qui répond à ces critères. La **classification**, elle, cherche à placer correctement dans la hiérarchie un nouveau frame. Ce dernier peut être un frame classe, auquel cas il s'agit de modifier la structure hiérarchique entre les frames pour y insérer le nouveau, ou bien un frame instance, et dans ce cas, il s'agit de le positionner dans la hiérarchie de frames classes existante.

2. Applications

Les frames représentent une structuration de la connaissance encore plus poussée que celles des réseaux sémantiques et des graphes conceptuels, puisque cette fois, la connaissance relative à un objet sémantique est concentrée dans un frame.

[Schreiber01], mais aussi [Hyvonen01] montrent, en utilisant un système à base de frames, que la formalisation de connaissances « domaine » dans une ontologie est une aide pour un processus d'annotation manuelle de photographies.

En outre, les systèmes à base de frames étant particulièrement adaptés à des problématiques de classement ([Drews93]), ils vont se révéler très intéressants pour notre étude. En effet, il s'agit, dans le cadre de notre méthodologie, tout d'abord de regrouper des segments en objets d'intérêt, puis de classer cet objet d'intérêt dans une catégorie d'objet sémantique.

F. Les travaux du web sémantique

Le web sémantique est un groupe de travail émanant du Consortium du World Wide Web (W3C), qui vise à rendre les ressources sur le web « *understandable* » (compréhensibles) pour les machines et pas

seulement « *readable* » (lisibles). Il s'agit donc de modéliser les documents du web d'après leur sémantique.

Les travaux présentés ici sont donc surtout orientés dans une optique de recherche de documents. Néanmoins, nous ne pouvons les ignorer, puisqu'ils interviennent juste après un processus d'indexation, thème de notre recherche. En outre, ils permettent eux aussi de formaliser des connaissances.

1. Les langages RDF, RDFS puis DAML+OIL

RDF (Resource description Framework) est un formalisme basé sur un modèle de graphes étiquetés et orientés. Il décrit une ressource, c'est-à-dire tout document identifié par un URI (Unique Ressource identifier), grâce à un triplet {Sujet, Prédicat, Objet} dans lequel « *Sujet* » désigne la ressource, « *Prédicat* » désigne le nom d'une propriété et « *Objet* » désigne la valeur de la propriété. En outre, les sujets et prédicats sont des URI, et les objets peuvent être des URI ou des littéraux.

Par exemple, la décrit la connaissance « *Nicolas est le créateur de la ressource <http://foo.perso/nicolas.htm> ».*

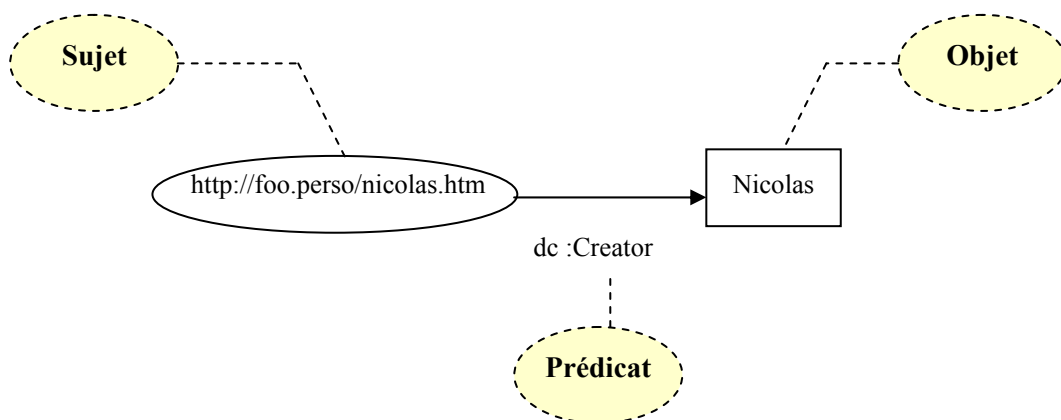


Figure 7- Représentation en RDF de « *Nicolas est le créateur de la ressource ...* »

RDF s'appuie sur une syntaxe XML pour représenter les ressources [Lassila99], mais ne dispose pas de vocabulaire propre. Ainsi, différents vocabulaires peuvent être utilisés pour décrire réellement les ressources. Par exemple, Dublin Core [DublinCore03] propose un ensemble de 15 propriétés, comme *Title* ou *Creator* (utilisé dans l'exemple de la Figure 7) afin de caractériser toute ressource.

A ce titre, RDFS (Resource Description Framework Schema) [Brickley03] permet de décrire des vocabulaires RDF avec une syntaxe RDF. Il est ainsi possible de spécifier le domaine des valeurs d'une propriété, ou bien de définir sur quelles ressources peut s'appliquer une propriété. Ceci peut être particulièrement utile pour formaliser des connaissances « domaine ».

[Lafon01] propose un système d'annotations manuelle et de recherche de photos en ligne par RDF, suivant des aspects métadonnées (date, auteur, lieu) mais aussi sémantique, en formalisant une connaissance domaine par RDFS.

Néanmoins, RDF et RDFS demeurent des langages de formalisation peu précis. RDFS ne peut imposer de cardinalités à une propriété par exemple. C'est pour pallier à ces imprécisions que les projets DAML et OIL se sont associés dans la création du langage DAML+OIL. OIL est un projet européen travaillant sur un langage de représentation d'ontologies basé sur les logiques de descriptions¹. DAML est pour sa part un projet américain de représentation d'objets de connaissances à l'aide de RDF.

DAML+OIL cherche à raffiner RDF, en introduisant des notions comme la disjonction de classes, les cardinalités des propriétés, l'égalité entre les classes, etc.

¹ Les logiques de description sont issues du langage KL-ONE, qui reprend la théorie des frames en ajoutant des principes relatifs aux réseaux sémantiques.

Ces langages sont actuellement toujours en cours de définition. Néanmoins, ils constituent une référence importante, en terme d'interopérabilité des représentations de connaissances sur le web et conditionnent donc les aspects liés aux échanges futures de connaissances : compatibilité et reconnaissances des formalismes, adéquation des vocabulaire utilisés, ré-utilisation de vocabulaires déjà existants...

2. Ontoseek

Présenté dans [Guarino99], Ontoseek est un système de recherche de pages web, sur le modèle des pages jaunes. Le contenu des pages ainsi que les requêtes sont modélisées par des graphes conceptuels, et la recherche est effectuée en utilisant l'opérateur de projection, comme expliqué dans la partie D.1.

Constatant que les termes des requêtes et d'indexation pouvait être ambigus hors contexte, les auteurs ont tenté de résoudre ce problème en utilisant l'ontologie lexicale WordNet. Celle-ci se compose d'éléments de base, appelés *synsets*, regroupant un ensemble de termes qui représentent un sens sémantique particulier d'un mot anglais. En outre, chaque synset est composé de terme appartenant à une même catégorie lexicale : nom, verbe, adjectif par exemple. Enfin, les *synsets* sont reliés entre eux par des relations de synonymie et de méronymie (terme plus spécifique).

Le Tableau 2 présente quelques *synsets* du mot « *light* ».

Synset	Définition	Exemple
1) light , visible light, visible radiation	(physics) electromagnetic radiation that can produce a visual sensation	"the light was filtered through a soft glass window"
5) light	an illuminated area	"he stepped into the light"
5) light	mental understanding as an enlightening experience	"he finally saw the light"; "can you shed light on this problem?"

Tableau 2- Exemples de *synsets* du mot « *light* »

En général, un mot de la langue courante est relié à plusieurs *synsets*. Ainsi, en indexant les documents par des *synsets*, il est possible de lever l'ambiguïté du vocabulaire.

Dans l'application présentée par les auteurs, les pages sont indexées manuellement et les requêtes sont construites soit en naviguant dans l'ontologie, soit en saisissant directement un *synset*. Néanmoins, WordNet n'étant qu'une base lexicale, il est impossible de vérifier la cohérence sémantique de certaines requêtes.

Ces travaux ont néanmoins permis de montrer que l'utilisation d'une base lexicale peut s'avérer très utiles pour lever des ambiguïtés terminologiques. [Wielinga01] présente un autre exemple de ce résultat, en utilisant un vocabulaire « domaine » lors d'une annotation manuelle d'images.

G. Conclusion sur l'état de l'art

Dans cette partie, nous avons montré qu'il est nécessaire de séparer connaissances et raisonnement lorsque la méthode de résolution d'un problème est a priori inconnue. En outre, après avoir dégagé une méthodologie de construction d'une ontologie, nous avons présenté différents formalismes de stockage. Dans la perspective de notre étude, il semble que les systèmes à base de frames soient particulièrement adaptés à nos besoins. Néanmoins, dans un premier temps, et dans un souci de rapidité et de facilité d'implantation, il pourra être envisagé d'utiliser des mécanismes de raisonnement non spécifiquement dédiés aux frames, comme des moteurs d'inférences par exemple.

III.Méthodologie : collaborer entre haut-niveau et bas-niveau

Notre approche pour traiter le processus d'indexation adopte les trois niveaux de l'image couramment admis : bas-niveau, niveau intermédiaire et haut-niveau. Tout d'abord, le bas-niveau repose sur l'obtention d'une image segmentée, dont les éléments, appelés **segments**, sont des régions homogènes en terme de couleur ou de texture par exemple.

Le niveau intermédiaire est ensuite obtenu en rassemblant plusieurs segments de l'image, à partir du bas-niveau, afin de former des objets présentant une cohérence sémantique, sans qu'il soit nécessaire de les nommer à ce stade. On parlera par la suite d'**objet d'intérêt**. On passe du bas-niveau au niveau intermédiaire par une phase d'analyse.

Enfin, le haut-niveau, qui reconnaît effectivement en tant qu'**objets sémantiques** (en les nommant) les objets identifiés au niveau intermédiaire. On passe du niveau intermédiaire au haut-niveau par une phase d'interprétation.

A. Présentation de la méthodologie

L'architecture proposée s'articule autour de ces trois niveaux, selon le processus collaboratif suivant (voir Figure 8): une première segmentation génère différents segments : zones homogènes de l'image, suivant les descripteurs associés. Dans notre cas, nous avons choisi des descripteurs de texture. Chacun des segments dispose de caractéristiques particulières comme sa forme, la valeur des descripteurs bas-niveau associés, des propriétés topologiques et directionnelles vis-à-vis de ses voisins, sa taille, etc. Ces informations, spécifiques à un segment dans l'image en cours de traitement sont regroupées dans une instance de connaissance bas-niveau. Puis une analyse de ces instances de connaissance bas-niveau a lieu, en exploitant les connaissances génériques « domaine ». Il est ainsi possible de dégager des objets d'intérêt, possédant également des propriétés topologiques, directionnelles, etc. qui sont cette fois regroupées dans des instances de connaissances « objets d'intérêt ». Enfin, une interprétation est effectuée, en exploitant les connaissances « domaine » et générant des instances de connaissances sémantiques. A chaque étape, des instances de connaissances, relatives à l'image en cours de traitement sont créées à partir des connaissances du niveau inférieur et des connaissances haut-niveau « domaine ».

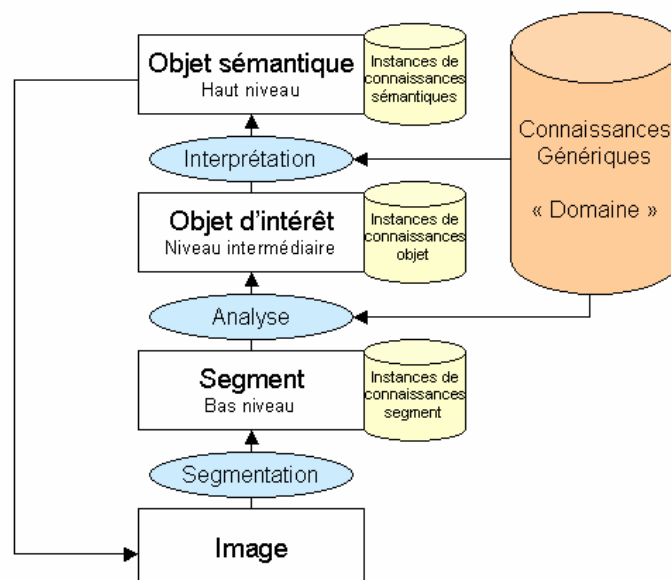


Figure 8- Méthodologie collaborative

Les segments qui n'ont pas passé la phase d'analyse et/ou d'interprétation sont renvoyés à la segmentation isolément. Ainsi, le système remet en cause l'étape de segmentation s'il ne parvient pas à conclure ou s'il parvient à des résultats erronés au niveau sémantique.

La démarche est collaborative dans le sens où les outils de segmentation, d'analyse et d'interprétation sont utilisés « au besoin ».

Etant donné la complexité du problème abordé, nous nous sommes focalisés sur les outils de segmentation et l'étape d'interprétation, dans une optique de démonstration de la faisabilité de notre approche.

B. Les traitements bas-niveau

Dans notre situation, les outils de segmentation devaient se conformer à deux contraintes majeures : être rapidement implantables avec peu de paramètres de réglage, et permettre une segmentation approximative mais suffisante des objets sémantiques.

Plusieurs options étaient possibles pour segmenter les images [Jain89]. Néanmoins, parmi celles-ci, le seuillage était difficile à mettre en œuvre, du fait de la grande variabilité des conditions de prises de vues. La couleur était de son côté, peu exploitable sur les images fournies, car beaucoup de celles-ci sont en noir et blanc. Les filtres d'extraction de contours (type Canny-Deriche) donnaient des résultats médiocres. En effet, les mauvaises conditions de prises de vues provoquaient un bruit important lors de l'extraction des contours.

Finalement, et compte-tenu du fait que les stèles présentent des zones très texturées (rosettes, inscriptions, cassures, etc...), des descripteurs de texture ont été choisis pour apporter une valeur sémantique aux données extraites.

1. Descripteurs de texture et clustering

Parmi les différentes méthodes basées sur la texture proposées par la littérature (matrice de co-occurrence de [Harralick73], méthodes fréquentielles de [Weszka76]), les filtres de Laws [Laws80] se conformaient bien aux contraintes imposées : simplicité, rapidité et pertinence des résultats pour le type d'images traité.

Les filtres directionnels de Laws sont au nombre de 25, issus des convolutions croisées de cinq filtres monodimensionnels (voir Tableau 3).

Chacun étant appliqué à l'image à traiter, ils fournissent 25 images de texture. Après un fenêtrage qui permet de calculer la variance de chaque pixel dans son voisinage, 25 images dites d'énergie de texture sont calculées. Enfin, en combinant 2 à 2 les images de direction privilégiée supplémentaires, il ne reste plus que 14 images d'énergie de texture.

L5	[1 4 6 4 1]	Level
E5	[-1 -2 0 2 1]	Edge
S5	[-1 0 2 0 1]	Spot
W5	[-1 2 0 -2 1]	Wave
R5	[1 -4 6 4 1]	Ripple

Tableau 3- Les 5 filtres monodimensionnel de Laws

A ce stade, un pixel de l'image originale est alors décrit dans un espace à 14 dimensions, 1 descripteur de Laws constituant 1 dimension. Dans le but d'obtenir une image segmentée, un procédé de clustering non supervisé (K-Means) est appliqué à l'image, afin de regrouper les pixels jugés similaires suivant les descripteurs de Laws.

Le principe du clustering non supervisé consiste à imposer un nombre N de clusters (regroupement) que l'on souhaite obtenir. Le procédé est initialisé en tirant au hasard N points A_i , pour représenter les barycentres initiaux de chacun des clusters. Un premier parcours de l'image permet d'associer chaque point P_j de l'image à un cluster C_k suivant la loi :

$$P_j \text{ appartient à } C_k \Leftrightarrow \text{distance}(P_j, A_k) = \text{Min}_i(\text{distance}(P_j, A_i))$$

La distance utilisée est la distance euclidienne dans l'espace à 14 dimensions.

Lorsque tous les points de l'image ont été affectés à un cluster, le barycentre A_i de chacun des clusters est recalculé. Tant qu'au moins un des barycentres est modifié par cette opération, le procédé est réitéré.

A l'issue du procédé de clustering, on dispose d'une image segmentée en N régions (ou clusters). L'avantage fondamental du clustering non supervisé est qu'il suffit de préciser au système le nombre de clusters à former, les classes étant ensuite construites dynamiquement.

2. Discussion et validation

Si l'implantation des descripteurs de Laws ne posent pas de problèmes en soi, il n'en va pas de même du clustering. En effet, une question importante surgit à ce stade : quel(s) descripteur(s) de Laws doit-on conserver : un seul, les 14, ou bien un échantillon choisi ?

La littérature fournit de nombreux cas d'utilisation des descripteurs de Laws. Néanmoins, dans la plupart des cas, un seul descripteur est exploité. Ainsi, [Chantler84] recense quatre descripteurs les plus fréquemment utilisés : L5E5, E5S5, R5R5 et L5S5¹.

Dans un souci de réduction du nombre de dimensions et de temps de calcul, nous avons choisi de conserver les descripteurs les plus discriminants pour nos données. Pour ceci, nous avons calculé, pour une image donnée, les corrélations croisées entre les 14 descripteurs. Ce calcul a été réalisé pour chaque image d'un lot de quatre, pris au hasard dans la base. Les résultats étant similaires, nous présentons dans le **Erreur ! Source du renvoi introuvable.**, les résultats des calculs de corrélations croisées pour une image. Les résultats sont normalisés, entre 0 et 1.

	L5E5	E5E5	L5S5	E5S5	S5S5	W5L5	W5E5	W5S5	W5W5	R5L5	R5E5	R5S5	R5W5	R5R5
L5E5		0.902	0.709	0.855	0.603	0.426	0.776	0.601	0.457	0.498	0.716	0.624	0.588	0.576
E5E5			0.862	0.952	0.767	0.576	0.887	0.758	0.611	0.656	0.823	0.770	0.740	0.724
L5S5				0.843	0.890	0.727	0.803	0.862	0.736	0.767	0.756	0.830	0.820	0.801
E5S5					0.796	0.614	0.964	0.816	0.668	0.720	0.911	0.842	0.808	0.794
S5S5						0.716	0.793	0.897	0.788	0.812	0.762	0.847	0.840	0.821
W5L5							0.617	0.701	0.778	0.662	0.587	0.671	0.656	0.654
W5E5								0.848	0.698	0.762	0.964	0.891	0.858	0.845
W5S5									0.775	0.860	0.838	0.920	0.911	0.894
W5W5										0.812	0.696	0.786	0.799	0.785
R5L5											0.774	0.862	0.885	0.888
R5E5												0.914	0.879	0.867
R5S5													0.964	0.946
R5W5														0.963
R5R5														

Tableau 4- Corrélations croisées entre les 14 descripteur de Laws. Les valeurs en rouge sont celles qui sont supérieures au seuil fixé (0.860), au dessus duquel 1 des 2 descripteurs est supprimé

En procédant par élimination des paramètres très corrélés (si la corrélation entre D_1 et D_2 est supérieure à un seuil, alors seul D_1 ou D_2 est conservé), nous observons effectivement que les quatre paramètres L5E5, E5S5, R5R5 et L5S5 couramment utilisés dans la littérature sont parmi les six paramètres les plus discriminants. Les deux autres sont W5L5 et W5W5.

Dans la suite, nous utilisons ces six descripteurs pour l'extraction des régions d'intérêt.

Afin de valider ce choix et de prouver que cette famille de descripteurs pouvait effectivement permettre d'isoler correctement des objets sémantiques de l'image, des tests quantitatifs ont alors été réalisés. Plus précisément, deux séries de tests ont été effectuées : la première afin d'isoler la stèle de son fond ; la seconde afin d'isoler un objet particulier : la paire de rosettes présente dans le corps de stèle. Pour chacune des séries, les traitements sont effectuée une fois manuellement et une fois avec nos outils de segmentation (texture et clustering). Trois mesures sont alors calculées : la *précision* mesure le nombre de pixels isolés par la segmentation automatique, par rapport au nombre de pixels isolés à la main. La mesure de *recouvrement* prend en compte les pixels faussement isolés par la

¹ Chaque fois que nous parlons d'un descripteur de Laws, comme L5E5 par exemple, nous désignons l'ensemble formé par les deux descripteurs symétriques L5E5 et E5L5.

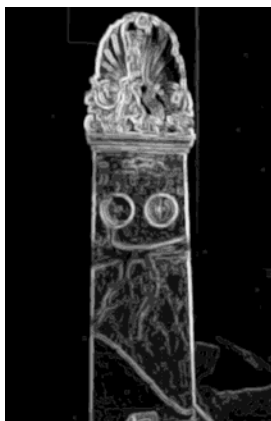
segmentation automatique. Enfin, la *non-détection* rend compte des pixels non isolés par notre procédé, alors qu'ils ont été isolés par la segmentation manuelle.

Le Tableau 5 résume les résultats de la première série. Deux approches ont été utilisées pour la segmentation automatique : tout d'abord l'extraction d'un ou plusieurs clusters choisi(s) par l'utilisateur. Cette approche présente néanmoins un inconvénient majeur : certaines textures étant présentes dans la stèle mais aussi dans le fond ne sont pas discriminantes. A l'issue de la segmentation, ces zones de texture, appartenant à deux objets sémantiques distincts, appartiennent donc au même cluster. Ceci explique certains résultats peu satisfaisants liés à cette méthode.

Stèle	Extraction de clusters			Labellisation par composantes connexes		
	Détection	Recouvrement	Non-détection	Détection	Recouvrement	Non-détection
GHW006	94	22	6	100	7	0
C074c1c	100	6	0	100	5	0
GHW041	98	3	1	100	3	0
IMG0037	82	25	18	100	6	0
GHW278¹	72	10	28	100	8	0
Moyenne	89	13	10.6	100	6	0

Tableau 5- Résultats de la segmentation pour isoler la stèle du fond (pourcentages)

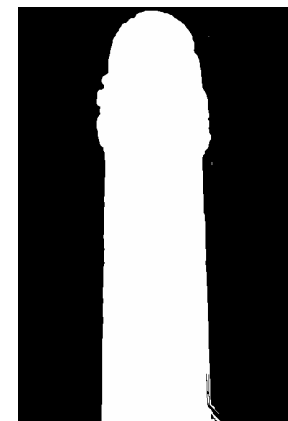
Pour pallier à cette limitation, une deuxième approche fut introduite : la labellisation par composantes connexes, à l'issue du clustering, puis la fusion manuelle des segments ainsi extraits en objets d'intérêt. L'opération de labellisation consiste, à partir des clusters, à extraire des segments connexes (Figure 9).



(a) Image de cluster. Les zones d'un même cluster sont affichées avec le même niveau de gris.



(b) Extraction puis fusion de 7 clusters parmi 8 pour extraire la stèle de son fond (en blanc). Les segments connexes mais appartenant au même cluster sont extraits (en bas, à droite)



(b) Labellisation par composantes connexes, puis fusion pour extraire la stèle (en blanc). Il est possible de dissocier des segments connexes appartenant au même cluster.

Figure 9- Extraction de clusters et labellisation par composantes connexes

Le Tableau 6 résume les résultats obtenus pour la deuxième série : extraction des paires de rosettes. Cette fois, seule la labellisation par composantes connexes a été utilisée.

1. Image ayant servi au calcul de corrélations croisées

Stèle	Labellisation par composantes connexes		
	Détection	Recouvrement	Non-détection
GHW006	74	8	25
GHW136	85	31	15
GHW278¹	88	26	12
IMG0037	71	15	29
Moyenne	80	20	15

Tableau 6- Résultats de la segmentation pour isoler les rosettes doubles (pourcentages)

Remarques :

- Le nombre N de clusters est arbitrairement fixé à 8 pour cette étude.
- Une partie des images utilisées pour les tests quantitatifs n'ont pas servi aux calculs de corrélations croisées, afin de valider le fait que la famille de descripteurs choisie soit pertinente pour toute la base d'images.

Ces études montrent que les outils de segmentation mis en œuvre sont suffisamment pertinents pour extraire les objets sémantiques des images. Ils ne permettent pas à eux seuls d'extraire chacun des objets séparément, mais ils fournissent une base solide pour les procédés d'analyse et d'interprétation qui suivront. Pour initier ceux-ci, il est tout d'abord nécessaire de modéliser et de stocker les connaissances haut-niveau.

C. La modélisation et le stockage des connaissances

Il s'agit d'identifier les concepts et les relations du domaine étudié, de les regrouper par leur sémantique, de donner à tous ces composants une définition, et enfin de les nommer.

Cette phase impose de mener une démarche collaborative entre les experts (ici ceux de la MOM) et l'équipe de modélisation (collecte d'informations, modélisation, contrôle) et ce, de manière itérative.

1. Modélisation

Concernant à ce stade l'ontologie domaine de la MOM, nous avons choisi de modéliser en tant que concepts les différents objets sémantiques constituant la stèle (corps, geison, rosette...). Comme tous ces objets ne sont pas du même type (une rosette est par exemple un élément de type « ornement »), d'autres concepts ont été introduits afin de regrouper les objets sémantiques entre eux. Dès lors, il devenait nécessaire de modéliser une relation de spécialisation entre les concepts, afin de pouvoir organiser les concepts entre eux, du plus général au plus spécialisé.

En outre, un autre concept s'est dégagé : celui de **région d'image**, désignant un segment d'image. Au sens strict, le concept « région d'image » n'appartient pas au domaine étudié. Il est pourtant inclus dans la modélisation puisque, d'après la première règle de [Ushold95], l'ontologie est construite dans un but donné, et que celui-ci impose de pouvoir manipuler le concept de « région ». De même, le concept d'**objet d'intérêt** est introduit, afin de décrire un ensemble de région représentant une cohérence sémantique.

Ainsi, les connaissances peuvent être distinguées entre celles relevant strictement du domaine étudié et celles plus orientées modélisation du problème. La Figure 10 présente cette séparation pour les premiers concepts de l'ontologie, organisés suivant une relation de spécialisation.

1. Image ayant servi au calcul de corrélations croisées

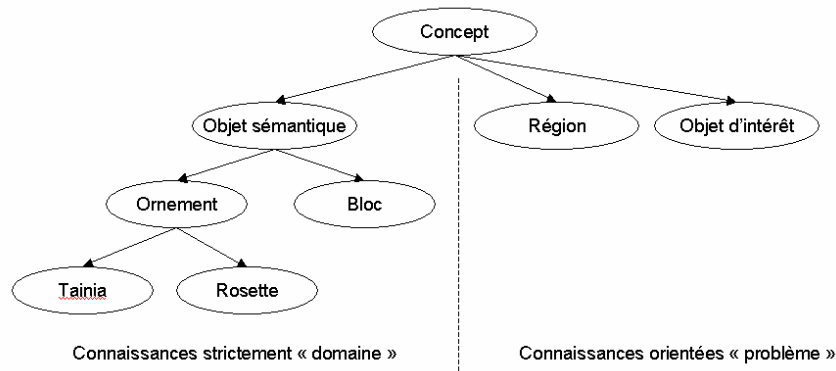


Figure 10- Les premiers concepts de l'ontologie (« domaine » et « problème »), organisés suivant la relation de spécialisation

A ce stade, l'ontologie se réduit à une série de concepts avec un seul type de relation entre ces concepts. Afin de pouvoir modéliser toutes les connaissances du domaine, il est nécessaire de créer d'autres relations entre les objets sémantiques, comme la relation de composition, d'être voisin à, etc. Ces relations, puisqu'elles relient deux ou plusieurs objets sémantiques appartiennent aux connaissances strictement domaine. De même, il est nécessaire d'introduire des relations topologiques et directionnelles entre des concepts *Régions* et *Objets d'intérêt*.

Dans le cadre de notre étude, nous avons modélisé les relations suivantes :

- Composition, par exemple : « un couronnement ogival est composé d'une rosette ».
- Adjacence entre deux segments ou deux objets d'intérêt ou encore deux objets sémantiques (en distinguant les adjacences haut, bas, droite et gauche) : « un couronnement est adjacent à un geison et est situé au dessus de lui ».
- Voisinage entre deux segments ou deux objets d'intérêt ou encore deux objets sémantiques, sans contact entre eux (en distinguant les cas nord, sud, est, ouest) : « une zone d'inscription est située au nord d'une paire de rosettes ».
- Inclusion directe entre deux segments ou deux objets d'intérêt.

Une fois les connaissances modélisées au niveau conceptuel, il s'agit de les stocker dans un formalisme adapté. A ce sujet, l'état de l'art réalisé et présenté en partie II nous a permis de nous orienter sur un système à base de frames.

2. Stockage dans un système à base de frames

Fondamentalement, et aussi bien du côté strictement « domaine » que du côté « problème », l'ontologie comprend deux parties : les concepts et les relations entre ces objets sémantiques.

Tous les concepts sont modélisés sous la forme de frames classes : en effet, ils correspondent à des connaissances génériques. Lorsque le modèle sera appliqué à une image particulière, (lors de l'interprétation par exemple), il sera alors instancié.

La relation de spécialisation trouve son équivalent exact dans la modélisation par frame, grâce à la notion d'héritage qui organise les frames suivant une relation de spécialisation. Cette hiérarchie permet de faire hériter à un frame les slots et leurs valeurs du frame parent. En outre, le frame peut être spécialisé par rapport à ses parents, en lui rajoutant des slots ou en ajoutant des facettes sur les slots hérités.

Ainsi, sur la figure précédente, le frame *Bloc* hérite de *Objet sémantique* par exemple.

La question se pose alors de modéliser les autres relations entre concepts. En effet, deux méthodes sont possibles : gérer la relation en tant que frame classe ou bien l'intégrer comme slot à un frame déjà existant (donc à un concept). La Figure 11 présente une illustration de ces deux méthodes pour la connaissance « un couronnement ogival est composé d'une rosette ».



(a) Modélisation de la relation de composition entre deux objets sémantiques par un slot.

(a) Modélisation de la relation de composition entre deux objets sémantiques par une frame classe en tant que telle.

Figure 11- Deux modélisations possibles pour une relation de composition entre objets sémantiques

Deux différences majeures apparaissent alors. Si la relation est modélisée via un slot, il sera moins aisé de contraindre certaines de ses propriétés comme la cardinalité par exemple. Ainsi, dans le cadre de notre exemple, un couronnement ogival est composé d'une rosette mais aussi d'autres éléments. Le slot composant du frame couronnement ogival doit donc être multi-valué et prendre ses valeurs parmi une liste de frame instances. Il n'est donc plus possible de préciser la cardinalité d'un type d'élément dans la relation. Cette limitation ne concerne pas la deuxième modélisation, puisque celle-ci crée une frame classe par relation de composition et permet ainsi de préciser, pour chaque relation de composition, les objets sémantiques impliqués ainsi que leur cardinalité.

Une variante consisterait à introduire un slot *composant* par type d'éléments impliqués dans la relation. Néanmoins, et cela demeure la différence fondamentale entre les deux modélisation, une relation modélisée par un slot perd sa sémantique de relation. En effet, rien ne distingue un slot d'un autre, sémantiquement parlant, si ce n'est leur nom. Ainsi, certaines classes pourront avoir certains slots représentant des relations et d'autres représentant des propriétés simples.

Il faut néanmoins noter que la modélisation des relations par des slots permet d'exploiter au maximum les avantages des modèles de frames, en regroupant les connaissances. Ce point est important en terme de simplicité des traitements par la suite, au niveau des raisonnements.

Au final, et compte-tenu de ces différences de modélisation, la solution retenue est la suivante : une relation entre plusieurs objets sémantiques (donc relevant des connaissances strictement « domaine ») est modélisée en tant que frame classe, car elle véhicule une sémantique importante au niveau du domaine étudié et doit alors être modélisée en tant que telle. Une relation entre un concept *Région* ou *Objet d'intérêt* et n'importe quel(s) autre(s) concept(s) est à l'inverse modélisée par un slot, dans un souci de simplification.

Ainsi, trois types de connaissances liées directement au domaine sont formalisées : la première consiste en une hiérarchie d'objets sémantiques selon la relation de spécialisation. La seconde consiste en une liste exhaustive des différentes relations structurelles pouvant exister entre ces objets. Enfin, la troisième consiste en l'introduction de connaissances « par défaut » par l'intermédiaire de facettes, contraignant les slots des frames concepts ou relations (voir à ce sujet la Figure 12).

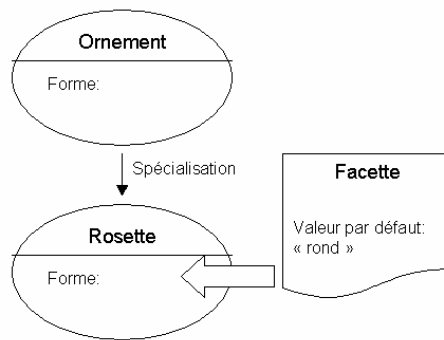


Figure 12- Modélisation de la connaissance par défaut « une rosette est ronde »

Ainsi, nous avons vu comment modéliser et formaliser la connaissance du domaine étudiée. Nous allons maintenant décrire comment cette connaissance peut être exploitée pour réaliser une analyse et une interprétation d'image, en s'appuyant sur les descriptions bas-niveau présentées dans la partie B.

D. Inférer des connaissances sémantiques

Le contexte est donc le suivant : les outils de segmentation, suivis d'une labellisation par composantes connexes donnent naissance à des instances de frame *Régions*. Chaque frame instance dispose de slots permettant de la décrire : taille relative à l'image, autres segments adjacents, segment dans lequel le segment en cours est éventuellement inclus, descripteurs de forme.

Un processus de raisonnement appliqué à ces instances et exploitant les connaissances de l'ontologie permet successivement l'instanciation de frames instance *Objet d'intérêt* et *Objet sémantique*, ainsi que des relations sémantiques correspondantes (voir la Figure 8, page14).

Pour l'instant, nous nous sommes focalisés sur la partie interprétation : étant donné des régions déjà fusionnées en objets d'intérêt ainsi que les instances correspondantes, chercher à identifier ces objets.

Pour permettre cette interprétation, un système à base de règles a été construit. Comme présenté dans la partie II.B, un tel système est composé de faits et de règles, constituant les connaissances, sur lesquelles s'applique un moteur d'inférences.

Les faits proviennent d'une part des connaissances liées aux objets sémantiques du domaine ainsi qu'à leurs relations (connaissances strictement « domaine »), et d'autre part des connaissances relatives aux objets d'intérêt extraits manuellement de l'image (modélisées dans l'ontologie par le frame *Objet d'intérêt*, connaissances orientées « problème »).

Plus précisément, aussi bien au niveau des connaissances « problème » que des connaissances « domaine », les faits sont issus aussi bien des frames classes correspondantes, et relatives aux connaissances génériques, que des frames instances, relatives aux connaissances spécifiques à l'image en cours de traitement.

Concernant maintenant les règles, on retrouve la distinction règles « domaine » et règles « problème ». Les premières manipulent des objets sémantiques et des relations associées pour déduire une sémantique à propos des objets d'intérêt. Néanmoins, si le système ne dispose que de ces connaissances et d'une série d'instances *Objet d'intérêt*, le procédé d'interprétation ne peut pas démarrer. Si, en revanche, il dispose en plus de la connaissance liée au fait qu'un objet d'intérêt désigne un objet sémantique, il peut alors initier le procédé à partir de ce fait. Les règles orientées « problèmes » permettent donc de fournir un point d'entrée au système, pour démarrer le processus d'interprétation. De telles règles permettent d'inférer une sémantique à partir de faits exclusivement orientés « problème ». A titre d'exemple, une telle règle est : « un objet au centre de l'image appartient à la stèle ».

La Figure 13 résume les différents connaissances manipulées pendant l'interprétation. Il est important de bien voir à ce stade que les connaissances sont constituées aussi bien par les faits que par les règles.

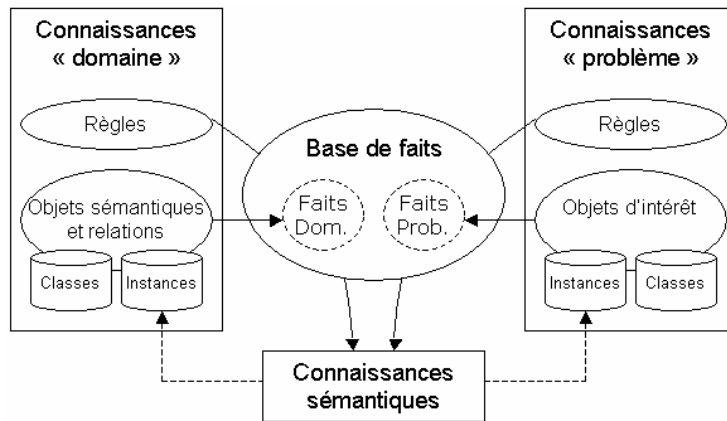


Figure 13- Connaissances (faits et règles) manipulés pendant l'interprétation

L'application d'une règle par le moteur d'inférence, permet d'acquérir une connaissance sémantique supplémentaire sur les faits. Cette connaissance, spécifique, est alors réinjectée dans le modèle, au niveau des frames instances, aussi bien dans les connaissances « problème » que dans les connaissances « domaine ».

Ainsi, si une instance *Objet d'intérêt* est reconnue comme objet sémantique par une règle, l'action déclenchée consistera à instancier l'objet sémantique correspondant, puis à appairer l'instance région avec l'instance de l'objet sémantique correspondant. En outre, les relations (*Composition, Adjacence*) entre les objets sémantiques seront également instanciées.

La Figure 14 présente un exemple de règle : partant de deux objets d'intérêt OI1 et OI2, adjacents suivant une configuration dessus/dessous, ainsi que de la connaissance générique de l'existence d'une relation entre deux objets sémantiques OS4 et OS5 ; si OI désigne OS4, alors on peut en déduire que OI2 désigne OS5. L'action consiste à instancier l'objet sémantique correspondant OS5 dans la base de connaissance, à appairer l'objet d'intérêt OI2 avec OS5, et enfin à instancier la relation entre les deux objets sémantiques.

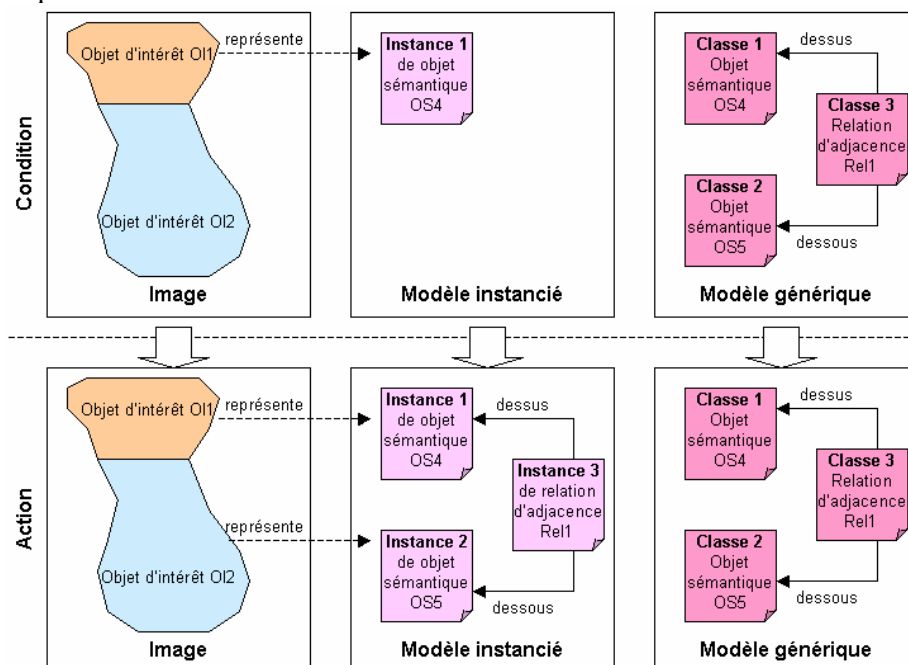


Figure 14- Exemple de règle « domaine » : identification d'objet sémantique à partir de connaissances génériques

Comme il n'est pas toujours possible pour une règle d'associer de manière non ambiguë un objet d'intérêt et un objet sémantique, le système a la possibilité dans ce cas d'élire plusieurs candidats à la représentation, pour indiquer, potentiellement, quel(s) objet sémantique(s) l'objet d'intérêt représente.

Un autre atout majeur de *Protégé2000*, réside dans le fait qu'il est possible d'ajouter facilement des fonctionnalités sous forme de plug-ins. Ces fonctionnalités supplémentaires relèvent aussi bien de la visualisation que de la manipulation des connaissances.

Ainsi, par exemple, le plug-in *Jess* permet d'introduire les connaissances stockées par *Protégé2000* dans une base de faits, en vue de l'application de règles par le moteur d'inférences *Jess*.

2. Raisonnement et inférences

Le raisonnement est effectué par le moteur d'inférence *Jess*. *Jess* est un système expert développé en Java, indépendant de *Protégé2000*. Le plug-in *Jess* de *Protégé2000* réalise un mapping en temps réel entre les objets de connaissances *Protégé2000* (les frames instances et classes) et les faits manipulés par *Jess*, autorisant ainsi une réelle collaboration entre raisonnement et connaissance.

Les règles sont, elles, stockées sous le formalisme lié à *Jess*.

Les règles implantées (une quinzaine environ) permettent pour l'instant de reconnaître un objet sémantique à partir d'objets d'intérêt. Ceux-ci sont extraits des images segmentées, en fusionnant manuellement les segments labellisés par composantes connexes. Puis, des descripteurs bas-niveau sont calculés automatiquement sur chacun des objets d'intérêt, comme la taille, l'excentricité, les paramètres du rectangle englobant, les graphes d'adjacence, etc. Ainsi, chaque instance objet d'intérêt OI possède les propriétés suivantes :

- Autres objets d'intérêt auxquels OI est adjacent, en distinguant les cas dessus, dessous, gauche et droite.
- Autres objets d'intérêt auxquels OI est voisin, sans être adjacent, en distinguant les cas nord, sud, ouest et est.
- Objet d'intérêt dans lequel OI est directement inclus
- Taille de OI relativement à l'image
- Forme de OI (reconnaissance manuelle).

A titre d'exemple, nous allons maintenant donner une illustration caractéristique de l'application du moteur d'inférence. Plusieurs jeux de tests ont été créés pour ceci et nous en présentons un simplifié ici. Une segmentation et un clustering ont été effectués sur une image de stèle et ont fourni la Figure 16. La Figure 17 illustre la situation initiale, après labellisation par composantes connexes, au moment du lancement de l'interprétation.

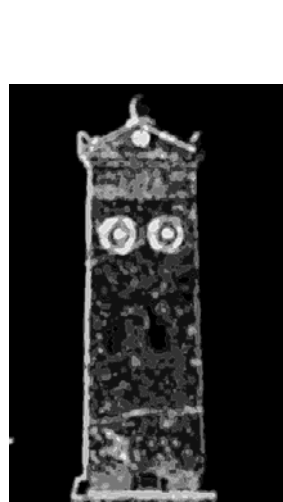


Figure 16- Image de stèle segmentée

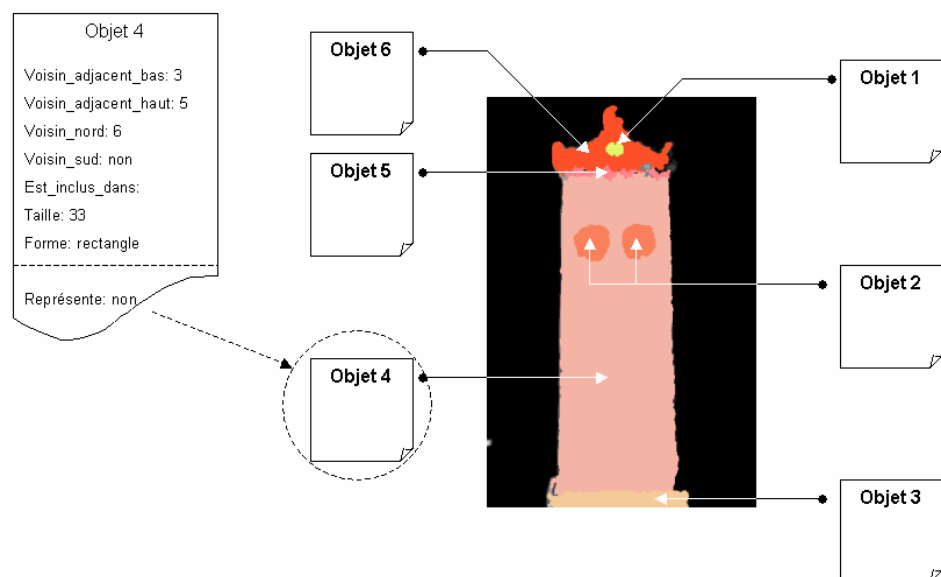


Figure 17- Exemple de situation initiale pour l'interprétation. Les objets sont extraits en fusionnant les segments issus de la labellisation par composantes connexe

Ensuite, les instances correspondantes (frames instance de *Objet* d'intérêt) sont créées dans *Protégé-2000*.

Une fois l'interprétation terminée, les résultats suivants sont disponibles, au niveau des objets sémantiques :

1. L'objet 4 a été reconnu comme *corps de stèle*, par application d'une règle « problème »
2. L'objet 5 a été reconnu comme un *geison*, parce que adjacent supérieur à un objet représentant un *corps de stèle*
3. L'objet 6 a été, dans un premier temps, reconnu comme couronnement, car adjacent supérieur à un objet représentant un *geison*. Comme *Couronnement* est une classe abstraite¹, il s'ensuit que toutes les classes filles de *Couronnement* sont potentiellement candidates pour être représentées par l'objet : *couronnement ogival*, *couronnement triangulaire*, *couronnement plat*.
4. Dans un deuxième temps, la forme étant un paramètre discriminant sur les trois candidats, la forme de l'objet 6 permet de conclure qu'il représente un *couronnement triangulaire*.
5. L'objet 3 est reconnu comme pouvant représenter un *tenon* ou un *socle*, grâce à sa relation d'adjacence avec l'objet 4, représentant un *corps de stèle*. Comme il n'existe pas de paramètres discriminant entre les deux classes candidates, le système ne conclut pas.
6. L'objet 2, puisque inclus dans l'objet 5, est reconnu comme étant candidat pour représenter : une *zone d'inscription*, un *naiskos*, un *champ de représentation*, un *tainia*, une *couronne*, un *hermes*, une *paire de rosettes*.
7. Comme l'objet 2 est rond, et que celle seule la *paire de rosette* est ronde dans la liste des candidats, le système conclut en affirmant que l'objet 2 représente une *paire de rosettes*.
8. L'objet 6 n'est pas reconnu : il n'a pas été introduit dans les connaissances « domaine ».

Remarquons que cette suite d'inférences est simplifiée par rapport à la réalité. En effet, il existe également des traitements visant à séparer les objets d'intérêt représentant le fond de l'image de ceux représentant des objets sémantiques par exemple. Ceci est effectué en exploitant des connaissances orientées « problème ».

Le système mis en place permet donc une identification d'objets si ces derniers possèdent une cohérence sémantique. Néanmoins, tous les objets ne peuvent pas être identifiés : en effet, il n'existe pas toujours de paramètre discriminant entre différents candidats.

Enfin, dans le cadre que nous nous sommes fixés pour le stage de DEA, les objets d'intérêt présentés au système sont extraits par un utilisateur, à partir des images segmentées. C'est sur la base de ces objets que le processus d'interprétation fonctionne, c'est-à-dire qu'implicitement, la connaissance que les objets étudiés représentent tous un objet sémantique est introduite dans le système. Néanmoins, dans la version finale de notre architecture, le système devra parvenir à générer lui-même les objets d'intérêt à partir des segments. Il peut alors arriver, par un enchaînement particulier de règles, que le système parvienne à des situations sémantiquement absurdes : par exemple, une zone de l'image mal segmentée, qui ne désigne pas un objet d'intérêt peut être par erreur désigné comme tel. Le processus d'interprétation peut alors aboutir à des situations incohérentes du type un objet *Couronnement* disposant d'un voisin supérieur appartenant à la stèle (*Objet sémantique*), alors qu'un couronnement est l'élément supérieur. Néanmoins, dans cet exemple, le fait qu'un couronnement ne dispose pas de voisin supérieur est modélisé dans l'ontologie. Le système est donc capable de signaler une incohérence sémantique. Une solution consisterait alors à exploiter ces situations absurdes, en remettant en cause les derniers résultats et en conservant dans une « mémoire » l'erreur commise.

¹ Une classe abstraite est une classe qui ne possède pas d'instance associée. Une telle classe modélise un concept générique, donc regroupant d'autres concepts, sans disposer d'une existence physique en tant que telle.

IV. Conclusion et perspectives

Nous avons présenté une méthodologie globale d'analyse et d'interprétation d'image, basée sur trois niveaux (bas-niveau, niveau intermédiaire et haut-niveau) selon un processus collaboratif, exploitant, à chaque étape, les connaissances du domaine dénoté par l'image. Pour ceci, nous avons insisté sur la nécessité de séparer connaissances et raisonnement, afin de pouvoir réellement utiliser les connaissances « domaine », sans nous contraindre dans un cadre procédural a priori. Dans cette optique, nous avons présenté une formalisation des connaissances dans un système à base de frames, particulièrement adapté à des problématiques de classement : étant donnés plusieurs segments d'image, comment les fusionner en objet d'intérêt, puis, comment interpréter ces objets d'intérêt en objets sémantiques.

En outre, cet aspect séparation entre connaissances et raisonnement nous permet de disposer d'un système modulable qui peut donc être mis à jour facilement. Ceci est particulièrement important puisque le processus de modélisation des connaissances est itératif. Ainsi, à chaque étape de modélisation en collaboration avec les experts du domaine, les nouvelles connaissances peuvent être rapidement intégrées dans un module de test pour vérifier leur bon fonctionnement lors de l'exploitation des connaissances.

Dans le cadre de la méthodologie globale, des descripteurs de texture, basés sur six filtres de Laws et une classification en régions par un clustering non supervisé ont été proposés. Des tests ont montré qu'ils étaient suffisamment pertinents pour extraire des objets sémantiques des images.

En outre, un moteur d'inférences a été utilisé sur des règles afin de vérifier qu'il était possible de réaliser une interprétation d'objets d'intérêt, extraits manuellement des images. Ainsi, notre étude a permis de valider la faisabilité de notre méthodologie

Nos recherches futures vont maintenant s'orienter sur deux axes : l'amélioration de l'efficacité des méthodes existantes d'une part et l'automatisation de toute la chaîne de traitements d'autre part.

Concernant le premier axe, une voie importante n'a pour l'instant pas été explorée. Il s'agit de l'utilisation d'outils de raisonnement spécifiques à la modélisation par frames, comme le filtrage (rechercher dans un ensemble de frames ceux qui correspondent à des critères donnés) et la classification (intégrer un nouveau frame dans une hiérarchie établie).

Concernant l'automatisation des traitements, une attention toute particulière sera portée au niveau intermédiaire, à travers l'ajout de règles permettant la fusion de régions en objets sémantiques d'une part, mais aussi en raffinant le modèle orienté « problème ».

Néanmoins, afin de ne pas dériver dans l'utilisation exclusive de règles orientées « problème », et donc peu réutilisables à d'autres banques d'images, une des pistes consistera à exploiter les interactions avec l'utilisateur. Ainsi, les actions de celui-ci visant à fusionner les segments d'image en objet sémantiques pourraient servir à un apprentissage par exemple.

Une autre orientation envisagée pour la phase d'analyse serait d'incorporer, en plus des connaissances « domaine », des connaissances « vision », relatives à la perception en tant que telle, suivant les principes de la vision Gestalt par exemple.

Enfin, toujours dans la perspective d'automatiser les traitements, nous envisageons d'intégrer notre application à Protégé sous la forme d'un plug-in à part entière.

Dans une perspective nettement plus à long terme, nous pensons également à traduire automatiquement les connaissances génériques « domaine » de l'ontologie (frames classes) sous forme de règles pour l'interprétation.

V. Remerciements

Mes remerciements les plus sincères vont à Bruno Tellez, pour m'avoir encadré durant mon stage de DEA. Ses conseils, son soutien furent des aides précieuses.

Je remercie Bruno Helly, Laurence Darnezin et Patrick Desfarges, de la MOM, pour toute leur participation en tant qu'expert du domaine modélisé.

Enfin, je remercie Catherine Roussey pour toute son aide au sujet de la modélisation de l'ontologie.

VI. Bibliographie

[Brickley03] D. Brickley, R. Guha “*RDF Vocabulary Description Language 1.0: RDF Schema*”, W3C, 23 jan 2003 URL: <http://www.w3.org/TR/rdf-schema/>

[Chantler84] M.J.Chantler. « *The effect of variation in illuminant direction on texture classification* », Ph.D. thesis, Dept. Computing and Electrical Engineering, Heriot-Watt University, 1994.

[Drews93] O. Marino Drews, « *Raisonnement classificatoire dans une représentation à objets multi-points de vue* », Thèse Informatique. Grenoble : Université J. Fourier, 1993, 275p.

[DublinCore03] Dublin Core Metadata Initiative. “*Dublin Core Metadata Element Set Version 1.1: Reference Description*”, 2003. <http://dublincore.org/documents/2003/02/04/dces/>

[Fox94] M. Fox, M. Gruninger. “Ontologies for Enterprise Integration”, in *Proceedings of the 2nd Conference on Cooperative Information Systems*, Toronto, Canada, 1994

[Fridman 00] N. Fridman Noy, R.W. Fergerson, and M.A. Musen, “The Knowledge Model of Protégé-2000: Combining Interoperability and Flexibility,” *Knowledge Engineering and Knowledge Management: 12th Int'l Conf. EKAW 2000*, Lecture Notes in Artificial Intelligence, vol. 1937, Springer-Verlag, Berlin, 2000, p. 17–32.

[Guarino99] N. Guarino, C. Masolo, G. Vetere. « *Ontoseek : Content-Based Access to the Web* », in *IEEE Intelligent Systems*, mai juin 1999, Vol. 14, N°3. p 70-80.

[Gruber93] T.R. Gruber, « A Translation approach to Portable Ontology Specification Knowledge Acquisition », *An International Journal of Knowledge Acquisition for Knowledge-based systems*, 1993, Vol. 5, no. 2.

[Haralick73] R.M. Haralick, K. Shanmugam, I. Dinstein. « Textural features for image classification », *IEEE Transactions on Systems, Man, & Cybernetics*, 1973, p. 611-621.

[Huertas96] A. Huertas, R. Nevatia. “Detecting Changes in Aerial Views of Man-Made Structures”. *Proceedings of the ARPA Image Understanding Workshop*. Palm Springs, California. 1996. p. 381-388.

[Jain89] A.K. Jain, « *Fundamentals of Digital Image Processing* », Englewood Cliffs, Prentice Hall, 1989. 569p.

[Kummert93] F. Kummert, H. Niemann, G. S. R. Prectel. « Control and explanation in a signal understanding environment », *Signal Processing*, 1993, Vol. 3, p. 111–145.

[Kunz97] D. Kunz, K. J. Schilling, T. Vogtle. « *A new approach for satellite image analysis by means of a semantic network* ». In W. Forstner and L. Plumer, editors, *Semantic Modeling*, p. 20-36, Basel. 1997

- [Lafon01] **Y. Lafon, B. Bos.** « Describing and Retrieving Photographs Using RDF and HTTP », *W3C Consortium*, 28 Sept. 2000, www.w3.org/TR/2000/NOTE-photo-rdf-20000928
- [Lassila99] **O. Lassila, R. Swick** “*Resource Description Framework (RDF) Model and Syntax Specification*”, W3C, 22 feb 1999 URL: <http://www.w3.org/TR/REC-rdf-syntax>
- [Laws80] **K. Laws.** « *Textured Image Segmentation* », Ph.D. Dissertation, University of Southern California, USA, 1980.
- [Li96] **J. Z. Li, M. T. Ozsü, D. Szafron.** « Spatial reasoning rules in multimedia management systems ». In *Proceedings of the International Conference on Multimedia Modeling MMM'96*, Toulouse, France, 1996
- [Mechkour95] **M. Mechkour, C. Berrut, Y. Chimarella,** « Using Conceptual Graphs Framework for Image Retrieval », Proc. Of the *MMM'95 (Multimedia Modeling) Conference*, Singapore, 1995, p. 127-142.
- [Mees96] **W. Mees.** “Automatic Image Interpretation for GIS updating”. *Fourth Euroconference on Geographical Information Systems*. Madrid. 1996. 7p.
- [Minsky75] **M. Minsky,** « A Framework for Representing Knowledge ». in *The Psychology of Computer Vision*, Edited by P. Winston. New-York : McGraw-Hill, 1975. p211-281.
- [Mulhem00] **P. Mulhem, D. Hong, J. Wu,** « Updating Labeling of Photographic Segmented Regions using Knowledge Representation Formalism and Dempster-Shafer Theory », *RIAO (International Conference on Computer Assisted Information Retrieval)*, France, 2000.
- [Niblack93] **W. Niblack, R. Barber, W. Equitz, et al.** « The QBIC Project : Querying Images by Content Using Color, Texture and Shape », in Proc. of *Storage and Retrieval for Image and Video Databases*, Volume 1908, pp. 173-187, Bellingham, WA, 1993.
- [Noy01a] **N. F. Noy, D. L. McGuinness.** « Ontology Development 101: A Guide to Creating Your First Ontology » *SMI Technical Report*, 2001.
- [Noy01b] **N. F. Noy, M. Sintek, S. Decker et al.** « Creating Semantic Web Contents with Protege-2000 ». *IEEE Intelligent Systems*, , 2001, Vol 16(2), p. 60-71.
- [Quillian68] **M. Quillian.** « *Semantic Memory* », in M. Minsky (ed.), *Semantic Information Processing*, MIT Press, Cambridge, MA, p. 227–270, 1968.
- [Roussey01] **C. Roussey.** « *Une méthode d'indexation sémantique adaptée au corpus multilingues* », Thèse Informatique. Lyon : Insa de Lyon, 2001, 196p.
- [Shortliffe76] **E. Shortliffe.** “*Computer-Based Medical Consultation: MYCIN*”, Elsevier, NY, 1976.
- [Smith96] **J.R. Smith, S.F. Chang,** « Tools and Techniques for Color Image Retrieval », in Proc. of *The Symposium on Electronic Imaging : Science and Technology Storage and Retrieval for Image and Video Databases IV*, pp. 426- 437, San Jose, CA, 1996
- [Sowa84] **J. Sowa.** “Conceptual Structures: information processing in mind and machine”, in *The System Programming Series*, Reading: Addison Wesley publishing Comp. 1984. 481p.
- [Ushold95] **M. Uschold , M. King,** « Towards a Methodology for Building Ontologies », in *Proceedings of the IJCAI '95 Conference: Workshop on Basic Ontological Issues in Knowledge Sharing*, Montreal, Québec, Canada, August, 20-25, 1995.

[Weszka76] J.S. Weszka, C.R. Dyer, A. Rosenfeld, « A comparative Study of Texture Measures for Terrain Classification » *IEEE Transactions on Systems, Man, & Cybernetics*, 1976 p. 269-285.

[Wielinga 01] B. J. Wielinga, A. T. Schreiber, J. Wielemaker et al. “From thesaurus to ontology”, *International Conference on Knowledge Capture*, Victoria, Canada, October 2001.