

## *Projet Orion*

*Environnements de résolution de problèmes pour des systèmes  
autonomes*

*Sophia Antipolis*

THÈME 3A



*R*apport  
*d'**A*ctivité

1999



## Table des matières

<b>1</b>	<b>Composition de l'équipe</b>	<b>3</b>
<b>2</b>	<b>Présentation et objectifs généraux</b>	<b>4</b>
<b>3</b>	<b>Fondements scientifiques</b>	<b>4</b>
3.1	Introduction . . . . .	4
3.2	Pilotage de programmes . . . . .	5
3.3	Interprétation automatique d'images . . . . .	8
<b>4</b>	<b>Domaines d'applications</b>	<b>10</b>
4.1	Panorama . . . . .	10
4.2	Imagerie astronomique . . . . .	10
4.3	Imagerie médicale . . . . .	11
4.4	Vidéosurveillance . . . . .	12
4.5	Reconnaissance de pollens . . . . .	14
<b>5</b>	<b>Logiciels</b>	<b>16</b>
5.1	Ocapi . . . . .	16
5.2	PEGASE . . . . .	16
5.3	VSIS . . . . .	16
<b>6</b>	<b>Résultats nouveaux</b>	<b>16</b>
6.1	Pilotage de programmes . . . . .	16
6.1.1	Modèle de tâche et vérification de systèmes de pilotage . . . . .	17
6.1.2	Le moteur MEDIA . . . . .	17
6.1.3	Pilotage de code de simulation numérique . . . . .	18
6.1.4	Plate-forme LAMA . . . . .	18
6.2	Interprétation automatique de séquences d'images . . . . .	19
6.2.1	Librairie d'échange de données en format XML . . . . .	20
6.2.2	Pilotage d'une chaîne de traitement d'images . . . . .	21
6.2.3	Logiciel d'aide à la calibration de caméra vidéo . . . . .	22
6.2.4	Modèle d'événements . . . . .	22
6.3	Reconnaissance semi-automatique de pollens . . . . .	26
6.3.1	Analyse de la forme de grains de pollen à partir de plusieurs plans images	28
<b>7</b>	<b>Contrats industriels (nationaux, européens et internationaux)</b>	<b>29</b>
<b>8</b>	<b>Actions régionales, nationales et internationales</b>	<b>29</b>
8.1	Actions régionales . . . . .	29
8.2	Actions nationales . . . . .	29
8.3	Actions financées par la Commission Européenne . . . . .	30
8.3.1	Projet ASTHMA . . . . .	30
8.4	Relations bilatérales internationales . . . . .	30

8.4.1	Europe	30
8.4.2	Afrique	30
<b>9</b>	<b>Diffusion de résultats</b>	<b>31</b>
9.1	Animation de la communauté scientifique	31
9.2	Enseignement universitaire	31
9.3	Thèses en cours	31
9.4	Participation à des colloques, séminaires, invitations	32
<b>10</b>	<b>Bibliographie</b>	<b>32</b>

# 1 Composition de l'équipe

## Responsable scientifique

Monique Thonnat [DR Inria]

## Responsable permanent

Sabine Moisan [CR Inria]

## Assistante de projet

Agnès Cortell [Assistante de projet Inria à temps partiel]

## Personnel Inria

Francis Avnaim [CR à temps partiel]

## Ingénieur expert

Nicolas Chleq [jusqu'en février]

Alain Boucher [à partir de mars]

## Personnel Bull

Robert Stahr [Ingénieur Bull Dyade]

## Chercheurs doctorants

Monica Crubézy [boursier INRIA, université de Nice-Sophia Antipolis, jusqu'en mars]

Nicolas Dey [boursier Région, université de Nice-Sophia Antipolis, à partir de novembre]

Maria del Mar Marcos [université Jaume I, Castellón, Espagne, jusqu'en février]

Jean-Christophe Noël [boursier Cifre CRIL INGENIERIE, université de Nice-Sophia Antipolis]

Nathanaël Rota [boursier INRIA Dyade, université de Nice-Sophia Antipolis]

## Collaborateurs extérieurs

Augustin Lux [Professeur, ENSIMAG, I.N.P.G]

Jean-Paul Rigault [Professeur, université de Nice-Sophia Antipolis]

## Stagiaires

Nicolas Dey [DEA SIC, Sup-Aero et université Paul Sabatier, Toulouse, d'avril à septembre]

Thomas Menguy [Supelec, de juillet à août]

Sebastien Wybo [ENS Cachan, de juillet à août]

## 2 Présentation et objectifs généraux

Notre objectif est de concevoir et de développer des techniques et des logiciels pour, d'une part, le **pilotage automatique de programmes** et, d'autre part, l'**interprétation automatique d'images**. Nous focalisons nos travaux sur l'étude des connaissances qui interviennent dans ces deux types de problèmes : connaissances sur les programmes et leur utilisation pour le pilotage automatique de programmes, et connaissances sur les objets et les scénarios à reconnaître pour l'interprétation automatique d'images. Thématiquement, Orion est une équipe pluridisciplinaire, à la frontière des domaines de la vision par ordinateur, des systèmes à base de connaissances et du génie logiciel. Les techniques que nous utilisons relèvent donc de ces trois disciplines. Les applications qui nous intéressent sont principalement du domaine de la vision par ordinateur. Plus précisément, notre objectif est de faciliter la construction de systèmes intelligents automatiques et adaptatifs, c'est-à-dire des systèmes incorporant explicitement une expertise (intelligents), fonctionnant sans intervention humaine pour les prises de décisions (automatiques) et ayant des capacités de réaction vis-à-vis des changements de leur environnement (adaptatifs). Parmi les applications qui nécessitent un fort degré d'autonomie, nous avons identifié, d'une part, les problèmes d'automatisation de l'utilisation de programmes, et, d'autre part, les problèmes d'interprétation automatique d'images. L'automatisation de l'utilisation de programmes nécessite de planifier les traitements et de contrôler l'exécution de codes de calcul. Par la suite nous utiliserons le terme **pilotage automatique de programmes** pour désigner ce type de problèmes. Dans de nombreuses applications, l'utilisateur ne peut pas interpréter les résultats des logiciels de traitements de données, soit parce qu'il n'en est pas capable (manque de compétence), soit parce qu'il n'en a pas le temps (saturation de ses capacités), soit enfin parce qu'il n'est pas présent (systèmes embarqués). Ceci correspond à un sous-problème de la perception, celui de l'interprétation des résultats des traitements d'images. Par la suite nous utiliserons le terme d'**interprétation automatique d'images** pour désigner ce type de problèmes.

## 3 Fondements scientifiques

### 3.1 Introduction

Afin de faciliter la construction de systèmes à base de connaissances (SBC), nous développons des moteurs indépendants d'une expertise particulière, mais cependant *dédiés* à une classe de problèmes. Les classes de problèmes étudiées sont le pilotage automatique de programmes et l'interprétation d'images. Le développement d'outils dédiés fournit des systèmes bien adaptés à l'expression des connaissances et largement utilisables dans différents domaines d'applications.

Pour construire de tels moteurs il est nécessaire de se baser sur une modélisation des connaissances et des méthodes de résolution de problèmes qui interviennent dans les classes de problèmes étudiées.

## 3.2 Pilotage de programmes

**Mots clés :** pilotage de programmes, planification, réutilisation de programmes, génie logiciel.

**Participants :** Monica Crubézy, Mar Marcos, Sabine Moisan, Jean-Paul Rigault, Monique Thonnat.

### Glossaire :

**Le pilotage de programmes** a pour but d'automatiser l'utilisation de logiciels complexes (bibliothèques de traitement d'images par exemple), grâce à des systèmes capables de planifier des traitements et d'en contrôler l'exécution.

**Résumé :** *Le pilotage de programmes est un domaine de recherches récent, pour lequel les techniques des systèmes à base de connaissances sont bien adaptées. L'objectif est d'automatiser l'utilisation de logiciels complexes. Nous étudions le problème de la modélisation des connaissances spécifiques au pilotage de programmes. Cette modélisation sert à définir, d'une part, des méthodes de vérification de bases de connaissances, et, d'autre part, des moteurs dédiés au pilotage de programmes. Enfin nous réalisons des prototypes de moteurs développés grâce à une plate-forme logicielle dont le but est de fournir un environnement commun à différents systèmes de pilotage.*

Le *pilotage de programmes* a pour but d'automatiser l'utilisation de logiciels complexes (bibliothèques de traitement d'images par exemple), grâce à des systèmes capables de planifier des traitements et d'en contrôler l'exécution. Le pilotage de programmes est un domaine de recherches récent. Un nombre croissant de travaux sont issus de nombreux domaines techniques et applicatifs. Ces activités de recherche sont souvent motivées par un domaine d'application particulier (tel que le traitement d'images, le traitement du signal ou le calcul scientifique).

Nous proposons une approche basée sur les techniques des systèmes à base de connaissances (SBC).

Un système à base de connaissances pour le pilotage de programmes est généralement constitué :

- d'une bibliothèque de programmes exécutables adaptés à un domaine d'application (par exemple : traitement d'images médicales),
- d'une base de connaissances recueillant le savoir-faire d'utilisation des programmes de la bibliothèque (choix, ordonnancement, paramétrage, exécution, évaluation des résultats, etc.),
- d'un moteur de pilotage qui choisit, ordonne et exécute les programmes de la bibliothèque en fonction de l'objectif et des données de l'utilisateur et des connaissances contenues dans la base de connaissances,
- d'une interface permettant à un utilisateur d'exprimer un objectif à atteindre sur des données particulières, de suivre l'exécution des traitements et de visualiser leurs résultats.

### Modélisation du pilotage

Afin de mieux comprendre la problématique du pilotage de programmes, d'améliorer la qualité de l'utilisation des programmes existants et d'élargir leurs domaines d'applications, les connaissances impliquées dans le processus de prise de décision de l'expert doivent être décrites de manière structurée et indépendante de l'implantation. Ceci peut être atteint en appliquant des techniques de modélisation des connaissances. Nous étudions dans quelle mesure les compétences sur le pilotage de programmes peuvent être représentées explicitement. Une description des connaissances de pilotage de programmes permet le transfert des compétences de l'expert qui a développé la bibliothèque vers des utilisateurs novices. Nous avons modélisé les concepts et les mécanismes du pilotage de programmes tout d'abord pour OCAP<sup>I</sup> [CT93], puis pour les moteurs récemment développés. Pour cela nous avons utilisé le modèle d'expertise de KADS. Ce travail de modélisation a été amélioré en se basant sur des techniques de réutilisation de composants logiciels (génie logiciel), des techniques de planification (intelligence artificielle), des systèmes existants de pilotage de programmes et sur notre expérience pratique sur des applications comme la détection d'obstacles dans des scènes routières, l'imagerie médicale et la classification de galaxies.

### Vérification de bases de connaissances

Le problème de la vérification de systèmes à base de connaissances de pilotage de programmes est important pour assurer leur robustesse. Cette exigence de robustesse est encore plus cruciale pour des systèmes autonomes. Nous avons choisi de nous appuyer non seulement sur les propriétés des modes de représentation des connaissances utilisés dans les SBC (schémas et règles), mais aussi sur le modèle sur lequel ces systèmes sont basés. Nous étudions des systèmes à base de connaissances construits grâce à des environnements de développement dédiés au pilotage et intégrant donc une modélisation fine de cette classe de problèmes. Cette approche nous offre un cadre approprié pour définir des méthodes de vérification qui prennent en compte les différents types de connaissance impliqués. Notre but est de fournir aux experts une aide pendant la construction d'une base de connaissances, en intégrant la vérification dans la méthodologie de développement du système à base de connaissances. L'application de méthodes de vérification classiques, fortement basées sur la syntaxe des modes de représentation des connaissances, doit être précédée d'une étude détaillée du rôle que les différents types de connaissances jouent dans les systèmes de pilotage. Ainsi, l'utilisation et la représentation des connaissances sont prises en compte de façon complémentaire pour la vérification. Ceci implique des vérifications syntaxiques, structurelles, aussi bien que sémantiques (sur le rôle des objets, par exemple). Étant donné une base de connaissances et le modèle du moteur de pilotage qui va l'utiliser, notre but n'est pas seulement de vérifier la consistance et la complétude de la base, mais aussi de vérifier l'adéquation de la connaissance par rapport à la façon dont le moteur va l'utiliser. Pour cela, nous avons entrepris la modélisation de différents moteurs de pilotage.

### Plate-forme de développement

La construction d'outils dédiés à une classe de problèmes particulière (comme le pilotage de programmes) offre deux avantages : tout d'abord focaliser les modèles de connaissances fournis par les outils sur les besoins de la classe de problèmes, et ensuite fournir des formalismes uni-

---

[CT93] V. CLÉMENT, M. THONNAT, « A Knowledge-Based Approach to Integration of Image Procedures Processing », *CVGIP: Image Understanding* 57, 2, mars 1993, p. 166–184.



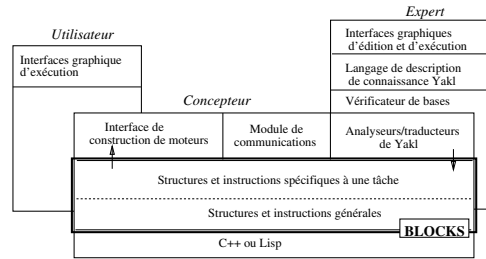


FIG. 1 – L’architecture de LAMA. La bibliothèque de composants réutilisables permet de construire des moteurs. La plate-forme comporte aussi le langage de description de bases de connaissances YAKL, des outils de vérification, des modes de développement sous Emacs et des interfaces graphiques.

fiés et communs aux bases de connaissances d’une même classe de problèmes. Nous souhaitons aller plus loin et faciliter aussi l’écriture des moteurs eux-mêmes, afin de pouvoir décrire une stratégie de pilotage à un niveau abstrait et de pouvoir facilement changer de stratégie selon les besoins. C’est dans ce but qu’a été développée la plate-forme LAMA <sup>[Moi98]</sup>, qui fournit un environnement commun à différents systèmes (de pilotage actuellement), comportant un langage de description et des facilités de vérification de bases de connaissances, ou des interfaces graphiques. Grâce à une telle plate-forme, il est possible de concevoir facilement différents moteurs. Une même base de connaissances peut être utilisée par différents moteurs, moyennant quelques modifications bien identifiées en fonction du moteur tout en gardant le même environnement de développement. La plate-forme va être étendue pour la construction de systèmes d’interprétation automatique.

La plate-forme (voir figure 1) comprend principalement une bibliothèque de composants réutilisables pour construire des moteurs (de pilotage de programmes en particulier), un langage de description de base de connaissances (YAKL), un serveur de messages (LAD), des interfaces graphiques, et des outils de vérification de base de connaissances.

*Bibliothèque de composants.* La bibliothèque BLOCKS (Basic Library Of Compoments for Knowledge-based Systems) est un « framework » offrant des composants réutilisables et adaptables pour les concepteurs de générateurs de SBC. BLOCKS comprend, d’une part, des structures représentant les concepts intervenant dans les SBC en général – et dans la tâche de pilotage de programmes en particulier –, et, d’autre part, des instructions nécessaires à la description des stratégies de raisonnement. Ceci facilite la construction de moteurs ou de variantes de moteurs en permettant de décrire leurs stratégies de raisonnement à un niveau abstrait, plus simplement et plus rapidement qu’avec un langage de programmation. De plus, les modifications des moteurs sont simplifiées en cas d’évolution des spécifications, des besoins ou des contraintes. Cette approche permet aussi une vision unifiée de différents moteurs et fournit une base commode de comparaison entre stratégies de raisonnement.

*Langage de description de bases de connaissances.* YAKL (Yet Another Knowledge base Language) permet de décrire le contenu d’une base de connaissances, sans préjuger du langage

[Moi98] S. MOISAN, *Une plate-forme pour une programmation par composants de systèmes à base de connaissances*, Habilitation à diriger les recherches, Université de Nice, avril 1998.

d'implémentation cible (Lisp ou C++). Ce langage sert à la fois de format de stockage commun à tous les moteurs de la plate-forme et de langage d'écriture ou de consultation de bases de connaissances. YAKL offre une syntaxe proche de la façon dont s'exprime l'expert pour décrire par exemple, les opérateurs, les buts, les règles de production qui constituent une base de connaissances en pilotage. Des outils de vérifications lui sont connectés et le code en langage d'implémentation cible est généré automatiquement.

Un mode Emacs associé a aussi été développé afin de fournir une aide syntaxique à l'écriture de bases de connaissances.

*Serveur de messages LAd.* Un système de pilotage de programmes est un système réparti, constitué de plusieurs processus applicatifs communiquant entre eux (interface, moteur, bibliothèque de programmes, etc). En particulier les programmes préexistent souvent au système de pilotage et doivent pouvoir continuer à être utilisés de manière autonome. De plus, un système de pilotage peut être amené à fonctionner comme un sous-système spécialisé d'un système plus général. Par exemple, un système de classification d'objets à partir d'images peut déléguer des tâches de traitement d'images à un système de pilotage [Oss96].

C'est pourquoi nous avons mis en place un mécanisme de communications assez général (LAd: LAMA communicAtion daemon) sur le principe des bus logiciels, basé sur des communications par sockets Unix TCP. Son architecture, de type client/serveur, centralise la gestion des communications grâce à un *serveur de messages*. Le serveur communique avec chaque processus client du système au moyen d'un *module générique d'émission/réception de messages* par client.

### 3.3 Interprétation automatique d'images

**Participants :** Nicolas Chleq, Monique Thonnat.

**Mots clés :** interprétation d'images, reconnaissance de formes, reconnaissance de scénarios, séquences d'images.

**Glossaire :**

**L'interprétation automatique d'images** consiste à attribuer une sémantique aux données en fonction de modèles prédéfinis. Cela correspond à un sous-problème de la perception, celui de l'interprétation automatique des résultats des traitements d'images.

**Résumé :** *L'interprétation automatique d'images est une problématique très difficile qui est la base de nombreux travaux en vision et aussi en intelligence artificielle. La difficulté dépend de la nature des entités à reconnaître et du type d'interprétation recherchée. Dans le projet Orion, nous nous intéressons à des images qui sont des entités numériques complexes spatiales et/ou temporelles (2D à 4D). Les résultats de l'interprétation peuvent être la reconnaissance d'objets physiques, d'événements, de situations ou de scénarios.*

---

[Oss96] J. OSSOLA, *Coopération de systèmes à base de connaissances pour l'analyse et la reconnaissance d'objets naturels complexes : application au classement de galaxies ou de zooplanctons*, thèse de doctorat, université de Nice-Sophia Antipolis, mai 1996.

L'interprétation automatique d'images est une problématique très difficile qui est la base de nombreux travaux en vision et aussi en intelligence artificielle. La difficulté dépend en premier lieu de la nature des entités à reconnaître. Il est plus simple de reconnaître des objets statiques et rigides en environnement manufacturé, que des comportements dynamiques de plusieurs objets non-rigides en environnement naturel. La difficulté dépend également du type d'interprétation recherchée. Le problème peut être, soit simplement d'étiqueter une entité bien déterminée que l'on peut mettre directement en correspondance avec des modèles, soit de détecter les entités, de les étiqueter et de vérifier leur cohérence (spatiale, temporelle, structurelle, etc).

Dans le projet Orion, nous nous intéressons à l'interprétation automatique de données spatiales et/ou temporelles (2D à 4D). Les résultats de l'interprétation peuvent être la reconnaissance d'objets physiques, d'événements, de situations ou de scénarios. Le problème d'interprétation se décompose en une étape de traitement d'images qui permet la détection des entités d'intérêt et une étape d'analyse de ces entités dépendante du but du système développé, qui peut être la classification d'objet ou l'analyse de comportement. La résolution d'un tel problème se révèle complexe en raison de la double nature des connaissances requises : en effet, l'extraction, à partir d'images d'entrée, des paramètres et des primitives essentiels à l'interprétation nécessite des appels à des programmes de traitement d'images ; de plus, l'interprétation proprement dite ne peut s'accomplir qu'avec la connaissance du domaine que possède un expert sur les entités à reconnaître ou à analyser. L'automatisation des appels aux programmes de traitement d'images est un problème de pilotage de programmes ; il est donc traité dans l'axe de recherche correspondant (cf. module 3.2). Les deux points présentés ici concernent les modèles proposés pour l'interprétation : d'une part, pour l'interprétation de séquences d'images allant jusqu'à la reconnaissance de scénarios, et, d'autre part, pour la reconnaissance d'objets complexes.

**Interprétation de séquences d'images :** l'interprétation de séquences d'images a pour objectif, pour ce qui nous concerne, de donner un sens à une scène décrivant des activités humaines, à partir d'images fournies par une caméra couleur, monoculaire et fixe. Cette interprétation de scène repose, en général, sur la coopération d'un module de traitement d'images, d'un module de suivi des objets mobiles et d'un module de reconnaissance du comportement des objets mobiles qui s'appuient sur une base de contexte (cf. figure 2). Il s'agit, pour le module de traitement d'images, de détecter les régions mobiles sur la séquence d'images. Le module de suivi associe les régions détectées afin de former et de suivre les objets mobiles. La tâche du module de reconnaissance des comportements consiste, grâce à des techniques d'intelligence artificielle, à identifier les objets suivis et à reconnaître leur comportement comme constitutif d'un ou plusieurs scénarios prédéfinis.

**Reconnaissance d'objets complexes :** la reconnaissance d'objets complexes a pour objectif, à partir de modèles sémantiques abstraits, la reconnaissance d'un objet non géométrique. Dans une première phase, les traitements d'images sont effectués sur l'image à interpréter ; ils permettent de détecter, d'isoler et de décrire l'objet grâce à des mesures numériques. À partir de ces mesures numériques, le système d'interprétation va ensuite classer l'objet à reconnaître parmi des hiérarchies de classes prédéfinies et structurées qui représentent les modèles sémantiques ; trois étapes récursives essentielles sont impliquées dans cette phase de classement : abstraction des données, mise en correspondance avec une classe prédéfinie, raffinement de

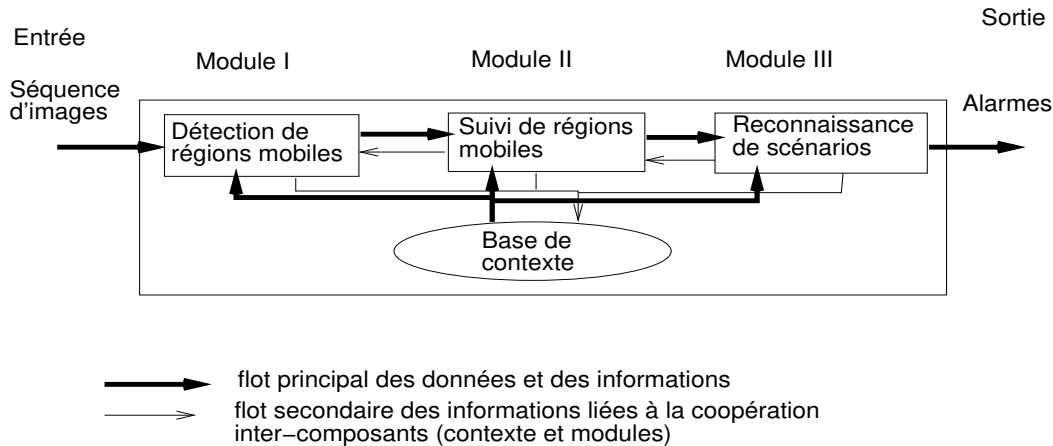


FIG. 2 – Architecture d'un système d'interprétation de séquences d'images.

la reconnaissance. Pendant la phase de classement, de nouvelles informations peuvent devoir être recherchées à partir de l'image. Notons que la réalisation de tels systèmes opérationnels requiert une somme considérable de travail pour le développement des bases de connaissances et des algorithmes de traitement d'images.

## 4 Domaines d'applications

### 4.1 Panorama

#### Résumé :

*Les applications abordées dans le projet ont un rôle essentiel pour mener à bien les travaux dans nos axes de recherches. Les grands domaines principalement représentés sont les suivants : l'astronomie, la santé et l'environnement, la vidéosurveillance et les transports. Les réalisations effectuées à l'aide de nos outils et méthodologies peuvent avoir deux finalités différentes : l'une, à caractère scientifique, est de permettre un apport dans d'autres disciplines, par exemple en astronomie pour la classification automatique de galaxies ; l'autre, à caractère industriel, est de permettre le développement de systèmes opérationnels comme la vidéosurveillance de stations de métro. Notons enfin que si l'imagerie est un domaine d'application privilégié du projet, des travaux applicatifs en calcul numérique, tel que le pilotage de codes de simulation numérique, ont également lieu (cf. module 6.1).*

### 4.2 Imagerie astronomique

*Résumé : L'automatisation complète de la description et du classement de galaxies selon leur type morphologique à partir d'images est une application historique*

du projet [Tho89,TB89]. Cette application pour laquelle nous disposons en interne de l'expertise aussi bien sur les traitements d'images de galaxies que sur les modèles théoriques de classification morphologique nous sert de support d'expérimentation pour valider nos modèles et nos logiciels en interprétation pour la reconnaissance d'objets complexes et en pilotage de programmes [TCO95,VTO97].

### 4.3 Imagerie médicale

Le pilotage de programmes de traitement (de séquences) d'images médicales pose des problèmes spécifiques. La complexité du domaine de l'imagerie médicale (données images variées et riches d'information, information imprécise et incomplète, paramètres multiples, objectifs méthodologiques et cliniques nombreux) conduit au développement de méthodes perfectionnées de traitement d'images médicales. Quoique performantes, ces méthodes sont diffusées de façon restreinte auprès des utilisateurs en routine clinique. Un système de pilotage de programmes de traitement d'images médicales, capable d'adapter les traitements d'images médicales à différents contextes, et permet de les mettre à la disposition d'utilisateurs non spécialistes comme les cliniciens. Parallèlement, la spécificité des besoins en pilotage de programmes de traitement d'images médicales fournit un cadre intéressant d'expérimentation d'un certain nombre d'extensions au pilotage de programmes. Nous travaillons, d'une part, sur la modélisation des connaissances (i.e. images, buts, traitements disponibles, besoins des utilisateurs), et, d'autre part, sur un mécanisme de planification et de contrôle d'exécution perfectionné pour respecter la démarche des experts. Enfin, une automatisation complète des opérations n'étant souvent ni possible ni souhaitable pour le traitement d'images médicales, nous nous intéressons au dialogue avec l'utilisateur clinicien durant la résolution de sa requête.

Nous avons travaillé sur deux applications, basées sur des approches différentes du traitement d'images médicales.

La première étude, menée en collaboration avec l'unité 494 de l'INSERM (La Pitié Salpêtrière, Paris), porte sur des programmes d'imagerie médicale fonctionnelle. Il s'agit de programmes d'Analyse Factorielle de Séquences d'Images Médicales (AFSIM [FBDP<sup>+</sup>92,BB<sup>+</sup>93]). La

- 
- [Tho89] M. THONNAT, *The World of Galaxies*, corwin, h. jr. and bottinelli, l. édition, Springer Verlag, 1989, ch. Toward an automatic classification of galaxies, p. 53–74.
- [TB89] M. THONNAT, A. BIJAOU, « Knowledge-based galaxy classification systems », in : *Knowledge-based systems in astronomy*, A. Heck et F. Murtagh (éditeurs), *Lecture Notes in Physics*, 329, Springer Verlag, 1989.
- [TCO95] M. THONNAT, V. CLÉMENT, J. OSSOLA, « Automatic Galaxy classification », *Astrophysical Letters and Communication* 31, 1-6, 1995, p. 65–72.
- [VTO97] R. VINCENT, M. THONNAT, J. OSSOLA, « Program Supervision for Automatic Galaxy Classification », in : *Proc. of the International Conference on Imaging Science, Systems, and Technology CISST'97*, juin 1997.
- [FBDP<sup>+</sup>92] F. FROUIN, J. BAZIN, M. DI PAOLA, O. JOLIVET, R. DI PAOLA, « FAMIS : A Software Package for Functional Feature Extraction from Biomedical Multidimensional Images », *Computerized Medical Imaging and Graphics* 16, 2, 1992, p. 81–91.
- [BB<sup>+</sup>93] H. BENALI, I. BUVAT *et al.*, « A statistical method for the determination of the optimal metric in factor analysis of medical image sequence (FAMIS) », *Physics in Medicine and Biology* 38, 1993, p. 1065–1080.

diversité des buts cliniques et des types d'images pour lesquels l'AFSIM est utilisée nécessite d'être modélisée au sein du système de pilotage par des concepts adaptés, intervenant à différentes étapes dans le raisonnement du moteur. De plus, cette diversité met en lumière différents niveaux de connaissances et de raisonnement, en particulier la distinction possible entre les connaissances génériques de pilotage de l'AFSIM et celles spécifiques à un type d'application de l'AFSIM. Enfin, les programmes de l'AFSIM soulèvent le problème de l'extraction des connaissances de pilotage qu'ils contiennent, couplé à des aspects de contrôle d'exécution non trivial. Une base de connaissances de pilotage de l'AFSIM (utilisée dans le but d'estimation de fonctions physiologiques) a été développée sur une application de référence : le « suivi de chimiothérapie sur ostéosarcome », qui traite des séquences temporelles d'images médicales en médecine nucléaire/IRM (cf. figure 3).

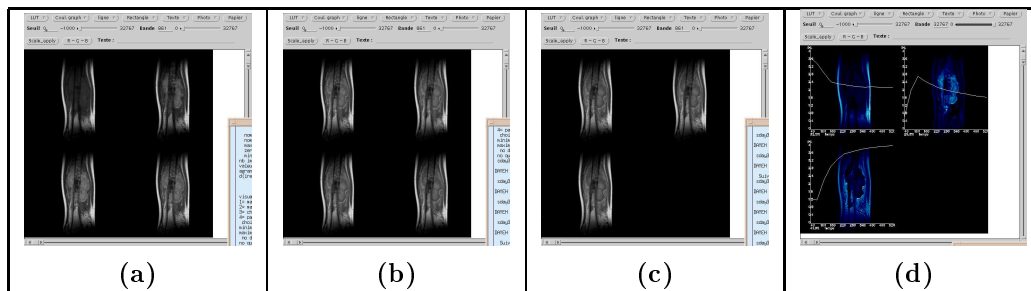


FIG. 3 – Séquence dynamique d'images IRM (genou tumoral en coupe sagittale) et résultats de l'AFSIM pour l'estimation des fonctions physiologiques sous-jacentes à un ostéosarcome du genou. (a), (b), (c) coupes sagittales du genou tumoral à un instant donné, (d) les 3 facteurs résultat superposés à leur localisation spatiale respective : facteur vasculaire rapide, facteur vasculaire lent et facteur d'accumulation dans les tissus interstitiels. (images fournies par l'INSERM-U66).

Notre seconde étude concerne une application de segmentation du cerveau à partir d'images IRM 3D, en collaboration avec Grégoire Malandain du projet Epidaure de l'Inria (cf. figure 4). Il s'agit de traitements d'images médicales basés sur les principes de la vision par ordinateur (morphologie mathématique essentiellement), visant à isoler le cerveau anatomique dans les images. Ici, les connaissances médicales utilisées par l'expert ne sont pas intégrées dans les programmes, mais interviennent à un haut niveau de raisonnement. L'intérêt principal de cette application réside dans la démarche de l'expert : elle est basée sur un mécanisme d'essai/erreur dû à la difficulté de déterminer les valeurs de paramètres appropriées et à la sensibilité de ceux-ci aux conditions d'acquisition. La réparation d'une solution prend ici toute son importance, et s'étend à la nécessité de mémoriser l'historique des différentes solutions choisies et des traitements subis par les images.

#### 4.4 Vidéosurveillance

Dans le domaine de la surveillance de locaux, le sentiment croissant d'insécurité parmi la population a conduit les sociétés privées et, dans une certaine mesure, les autorités publiques, à intégrer des systèmes de sécurité pour protéger leurs équipements ou leurs intérêts commer-

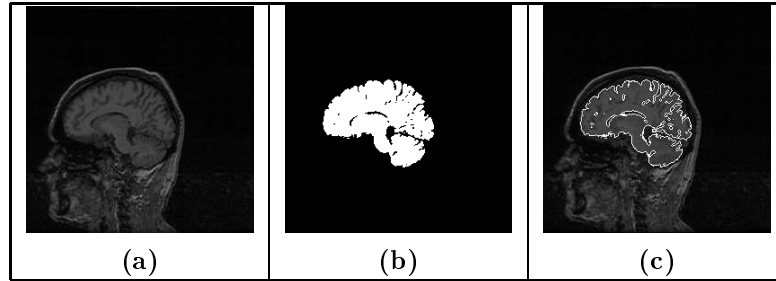


FIG. 4 – Exemple de segmentation d'images du cerveau : (a) IRM du cerveau initiale, (b) résultat de la segmentation du cerveau, (c) superposition du résultat sur l'image initiale pour une meilleure visualisation.

ciaux. Pour la sécurité des lieux publics, les techniques de surveillance par caméras sont de plus en plus utilisées, mais la multiplication du nombre de caméras entraîne une saturation des moyens de transmission et des moyens d'analyse par les contrôleurs (il est difficile de surveiller simultanément des centaines d'écrans). Par exemple, il y a actuellement plus de 1000 caméras pour surveiller le réseau du métro de Bruxelles. Dans le cadre de nos travaux en interprétation automatique d'images vidéo, nous étudions depuis 1994 le problème du développement d'un système automatisé d'assistance aux opérateurs de vidéosurveillance.

L'objectif d'un tel système est de fonctionner comme un filtre ayant pour fonction de trier les scènes susceptibles d'être intéressantes pour l'opérateur humain. Construit sur la coopération d'un module de traitement d'images et d'un module d'interprétation basé sur des techniques d'intelligence artificielle, ce système doit reconnaître automatiquement différents scénarios présentant un intérêt quelconque, afin de les présenter aux opérateurs. Ces travaux ont été effectués avec les partenaires académiques et industriels des projets européens Esprit Passwords, AVS-PV et AVS-RTPW. Une première gamme d'applications très simples pour la surveillance de nuit de l'intérieur d'un supermarché (AUCHAN) a montré la faisabilité de cette approche. Une deuxième gamme d'applications de type « surveillance de parking » où l'angle de vue assez large permet de voir de nombreux objets de nature distincte (voitures, piétons, chariots, etc.) dans un environnement changeant (éclairage, voitures en stationnement, arbres agités par le vent, etc.) a été étudiée (voir figure 5). Cette gamme d'applications a permis d'expérimenter diverses méthodes de récupération du suivi, d'analyse de trajectoire, de reconnaissance de cas typiques (occultation, formation et séparation de groupes, etc.). Nous avons fait porter nos efforts depuis 1997 sur l'étude des techniques de vidéosurveillance dans le domaine du transport nécessitant l'analyse et la reconnaissance de groupes de personnes à partir de vue latérales basses prises dans des stations de métro (métros de Nuremberg, Bruxelles et Charleroi). Plus récemment, en coopération avec Bull dans le cadre de l'action Dyade Telescope, nous travaillons sur la conception d'un système intelligent de vidéosurveillance indépendamment d'une application particulière, où les principales contraintes sont l'utilisation d'une seule caméra fixe et la possibilité de décrire les situations à reconnaître liées à l'application dans des modèles de scénarios indépendants du système de reconnaissance.

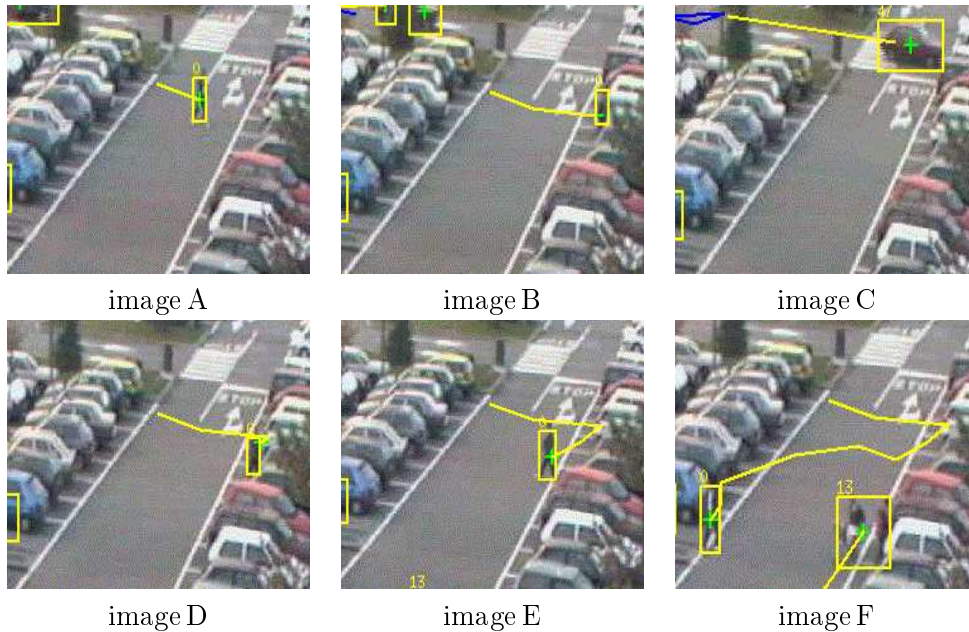


FIG. 5 – Exemple de scène de parking où un individu zigzague dans l’allée.

#### 4.5 Reconnaissance de pollens

**Résumé :** *Dans le domaine de l’environnement et de la santé, le projet Orion s’intéresse à l’automatisation de la détection et de la reconnaissance de pollens anémophiles, ce qui permettra de traiter des données beaucoup plus nombreuses et dans des délais plus courts que ce qui est réalisé actuellement manuellement par les analystes en palynologie. Pour cela, nous mettons en œuvre des méthodes de reconnaissance d’objets complexes faisant appel au traitement d’images, à la reconnaissance de formes et aux systèmes à base de connaissances.*

Nos travaux prennent place dans un projet de santé publique avec des objectifs épidémiologiques et médicaux. Les objectifs épidémiologiques sont ici de mettre en évidence le rôle et l’importance des différents facteurs et cofacteurs de pollution de l’air dans l’augmentation de la prévalence des maladies allergiques ainsi que d’intégrer un réseau de surveillance des pollens anémophiles dans les réseaux existant de surveillance de la qualité de l’air.

Du point de vue médical, il s’agit de réduire la fréquence et l’intensité des manifestations cliniques de l’allergie aux pollens, et d’assurer des actions de prédiction, d’information et de prévention médicale. Afin d’atteindre ces objectifs, les mesures de détection et de reconnaissance des différents types de pollens doivent être effectuées, d’une part, sur un réseau plus dense (il y a seulement 38 points de prélèvements en France) et, d’autre part, de manière beaucoup plus rapide (les résultats de mesures sont diffusés entre 8 et 15 jours après les prélèvements). Pour ces deux raisons, l’automatisation de la détection et de la reconnaissance des grains de pollens est souhaitée.

De tels travaux n’ont commencé que très récemment dans la communauté scientifique ; ils



sont basés sur l'utilisation de données images numériques, acquises à partir de lames, qui sont ensuite traitées par ordinateur. Deux directions principales sont étudiées : le comptage de tous les grains de pollen présents sur une lame dans le but d'avoir un compte global de la quantité de pollens, et la reconnaissance individuelle de chaque grain pour avoir une quantité précise pour chaque type de pollen. En ce qui concerne l'estimation globale de la quantité de pollens, son automatisa-tion a commencé à être étudiée en utilisant des techniques de traitement d'images, à partir de 1995, que ce soit au Japon [KTAN95] ou en France [TBA+97]. Toutefois, l'intérêt d'un système d'analyse automatique de pollens réside dans sa capacité à reconnaître le type de chaque grain de pollen ; c'est à ce problème qu'Orion s'est intéressé dès la fin de l'année 1996 [Maz97].

À cause de la complexité des différentes formes de pollens, nous étudions une approche prenant en compte la connaissance des palynologues ; pour cela, nous nous appuyons sur la coopération entre deux types d'outils : d'une part, des algorithmes de traitement d'images et de reconnaissance de formes et, d'autre part, des outils d'intelligence artificielle sous forme de systèmes à base de connaissances. Les algorithmes de traitement d'images et de reconnaissance de formes doivent permettre d'isoler les grains de pollens, et de réaliser les mesures numériques sur les grains susceptibles d'être des pollens. Les systèmes à base de connaissances doivent permettre de reconnaître les différents types de pollens en s'appuyant sur une modélisation taxonomique des espèces, et éventuellement, à terme de piloter les algorithmes pour optimiser les traitements. Notons que nous avons déjà utilisé cette approche dans d'autres domaines d'applications pour la reconnaissance d'objets naturels complexes tels que les galaxies, les zooplanctons [TG92,Oss96] ou les foraminifères [LSMTB96,LTB94].

Après avoir mené en 1997 en collaboration avec les palynologues du CEMBREU et le service de pneumologie du CHU de Nice, une étude de faisabilité sur ce sujet montrant la nécessité de travailler en 3D, nous avons monté en 1998 le projet européen ASTHMA dont le but est de prévoir et de fournir des informations sur la qualité de l'air pour les personnes sensibles

- 
- [KTAN95] S. KAWASHIMA, Y. TAKAHASHI, S. AIKAWA, T. NAGOYA, « An attempt of applying the image processing for the automatic estimation of sampled airborne pollen », *Japanese Journal of Allergy* 44, 9, 1995, p. 1150–1158.
- [TBA+97] R. TOMCZAK, P. BONTON, C. AUROYER, D. CAILLAUD, C. ROUQUET, « Traitement d'images et reconnaissance des formes appliquées à la mesure des taux de pollens dans l'air », *in : Journées Automatique, Agriculture et Agro-alimentaire*, Clermont-Ferrand, France, 1997.
- [Maz97] M. MAZIÈRE, « Étude de faisabilité pour la reconnaissance automatique de grains de pollen », 1997, Rapport de DEA, université de Nice-Sophia Antipolis.
- [TG92] M. THONNAT, M. GANDELIN, « Un système expert pour la description et le classement automatique de zooplanctons à partir d'images monoculaires », *Traitement du signal, spécial I.A* 9, 5, novembre 1992, p. 373–387.
- [Oss96] J. OSSOLA, *Coopération de systèmes à base de connaissances pour l'analyse et la reconnaissance d'objets naturels complexes : application au classement de galaxies ou de zooplanctons*, thèse de doctorat, université de Nice-Sophia Antipolis, mai 1996.
- [LSMTB96] S. LIU, P. SAINT-MARC, M. THONNAT, M. BERTHOD, « Feasibility study of automatic identification of planktonic foraminifera by computer vision », *Journal of foraminiferal research* 26, 2, avril 1996, p. 113–123.
- [LTB94] S. LIU, M. THONNAT, M. BERTHOD, « Automatic Classification of Planktonic Foraminifera by a Knowledge-based System », *in : The Tenth Conference on Artificial Intelligence for Applications*, IEEE Computer Society Press, p. 358–364, San Antonio, Texas, mars 1994.

aux aéroallergènes (pollens et autres poussières présents dans l'air). Dans ce cadre, le projet Orion s'intègre dans l'étude d'un système de reconnaissance semi-automatique de pollens et est responsable de la reconnaissance 3D de grains de pollen (cf. module 8.3)

## 5 Logiciels

### 5.1 Ocap

**Résumé :** *Jusqu'en 1996 le projet Orion a continué à développer et à diffuser le moteur de pilotage de programmes OCAP version 2.0. Les utilisateurs appartiennent au domaine industriel (NOESIS, Geoimage, CEA/CESTA) ou académique (Observatoire de Nice, Observatoire de Paris à Meudon, Université de Maryland).*

### 5.2 PEGASE

**Résumé :** *Depuis septembre 1996 le projet Orion diffuse un nouveau moteur de pilotage de programmes PEGASE, basé sur la plate-forme LAMA. PEGASE a été utilisé à l'université de Maryland et chez Genset.*

### 5.3 VSIS

**Résumé :** *Dans le cadre de l'action DYADE Telescope nous avons développé en C le logiciel VSIS (VisualSurveillance Intelligent Software). Ce logiciel comprend des modules de traitement d'image pour la détection et le suivi de personnes ainsi que des modules d'interprétation du comportement des personnes détectées et suivies. Les entrées du logiciel sont essentiellement une séquence d'images vidéo, une description de la scène vide observée par la caméra et un ensemble de modèles de comportements à reconnaître. En sortie, si un comportement intéressant correspondant à un des modèles prédéfinis est reconnu, le logiciel transmet un message d'alarme typé. VSIS permet l'analyse et l'interprétation de séquences d'images prises par une caméra monoculaire fixe.*

## 6 Résultats nouveaux

### 6.1 Pilotage de programmes

**Participants :** Monica Crubézy, Mar Marcos, Sabine Moisan, Jean-Paul Rigault, Monique Thonnat.

**Résumé :** *Cette année, trois thèses en pilotage de programmes se sont terminées, une sur la vérification de systèmes de pilotage, une sur le pilotage de programmes de traitement d'images médicales et une sur le pilotage de codes de simulation numérique. De plus, nous avons poursuivi le développement de la plate-forme LAMA en C++.*

### 6.1.1 Modèle de tâche et vérification de systèmes de pilotage

**Participants :** Mar Marcos, Sabine Moisan.

La thèse de M. Marcos [7] a finalisé la modélisation de systèmes de pilotage de programmes dans le but d'identifier les propriétés de la connaissance intéressantes pour la vérification des bases de connaissances et des moteurs. Les modèles des moteurs PEGASE, PULSAR et MEDIA ont permis d'identifier les hypothèses sur la connaissance du domaine de chaque moteur. Nous avons utilisé ces hypothèses pour déterminer les propriétés à vérifier sur les bases de connaissances et pour implanter un module de vérification, dit « à base de modèle », pour PEGASE. Les conclusions de ce travail sont que les techniques de génie cognitif peuvent se combiner avec les techniques classiques en vérification pour obtenir des outils de vérifications mieux adaptés aux besoins des utilisateurs et à la tâche réalisée par le système à base de connaissances. La construction et l'adaptation de modules de vérification de bases de connaissances est un aspect important de la plate-forme LAMA. C'est pourquoi une « boîte à outil » (*framework*) pour la vérification, permettant la construction de tels modules, en fonction des besoins de différents moteurs, est en cours d'intégration dans la plate-forme.

De plus, nous avons en parallèle identifié et formalisé certains aspects de la construction des moteurs de pilotage. Pour cela nous avons utilisé KIV (Karlsruhe Interactive Verifier) qui est un outil de vérification de programmes traditionnels, basé sur une formalisation logique. Ce type de vérification est peu abordé en général, car les travaux s'intéressent souvent à un seul moteur donné, dont le comportement est supposé correct.

L'approche « à base de modèle » permet de mieux guider l'acquisition des connaissances pour un expert, ou les étapes de construction d'un moteur.

### 6.1.2 Le moteur MEDIA

**Participants :** Monica Crubézy, Sabine Moisan.

Le moteur MEDIA (Medical Image processing Assistant), a été conçu pour répondre aux spécificités du domaine du traitement d'images médicales. Ce moteur a été implanté au cours de la thèse de M. Crubézy [6] en utilisant la plate-forme LAMA. Ce moteur a été testé sur deux applications en imagerie médicale, où il apporte une aide intéressante à l'utilisation de traitements complexes et peu diffusés, du fait de leur complexité (voir 4.3).

L'expérience de MEDIA a en outre servi de validation pour la réutilisabilité et l'extensibilité des composants de la bibliothèque BLOCKS. La réalisation de MEDIA a en effet réutilisé près de 90% des composants de la bibliothèque utilisés dans PEGASE. Des extensions ont été réalisées pour les besoins spécifiques de ce moteur, et intégrées dans la bibliothèque, notamment en ce qui concerne la prise en compte de conditions non bloquantes et le raffinement des instructions de planification. L'élaboration de ce nouveau moteur a aussi participé à l'enrichissement de la plate-forme, car ce travail a permis la mise en place du gestionnaire de communications de LAMA : LAD.

Même si l'objectif initial était de répondre aux besoins du pilotage en imagerie médicale, il s'avère que les capacités de MEDIA en termes de planification hybride, de représentation des connaissances riche et de contrôle de programmes distants, sont applicables à d'autres

domaines que celui du traitement d'images médicales.

### 6.1.3 Pilotage de code de simulation numérique

**Participants** : Jean-Christophe Noël, Monique Thonnat.

Afin d'étudier dans quelle mesure nos travaux sur le pilotage de programmes peuvent s'appliquer à des codes de simulation numérique nous coopérons avec le laboratoire Cesta du CEA à Bordeaux. L'objectif est, d'une part, de capitaliser les connaissances sur l'utilisation de codes de simulation numérique et, d'autre part, de mettre en œuvre cette capitalisation dans un système opérationnel de pilotage. La capitalisation des connaissances implique que la représentation choisie soit dotée d'une expressivité riche et soit également facilement consultable par les experts en simulation numérique. Comme ces connaissances doivent être utilisées pour piloter effectivement des codes, la représentation de ces connaissances doit être très précise et complète. Pour concilier les aspects capitalisation et résolution de problèmes, nous avons conçu un nouveau système appelé Licorne comprenant un langage et un moteur. Ce langage de représentation des connaissances est essentiellement basé sur des graphes. Le langage permet d'exprimer l'organisation des connaissances grâce à la notion de groupement de noeuds de graphe, à la notion de schéma avec des noeuds et des relations ainsi que la notion de modèle de schémas. Des composants génériques, qui sont des modèles de schémas opérationnels, ont été développés. Parmi ceux-ci, une famille de composants génériques modélise un parcours itératif intégrant explicitement la notion de convergence. Il est aussi possible d'exprimer dans ce langage certains concepts du pilotage qui sont bien adaptés aux codes de simulation numérique, tels que les aspects liés au réglage de paramètres (initialisation, ajustement de paramètres), ainsi que la notion de validation de résultats.

Deux applications Licorne ont été développées, l'une dans le domaine de l'infrarouge pour tester les aspects réglage de code grâce à l'analyse par perturbations, l'autre dans le domaine du flambage pour spécifier et automatiser en mode coopératif un parcours itératif d'exploration des valeurs propres d'un spectre discret. Les deux bases de connaissances Licorne de ces applications ont permis le pilotage effectif de codes d'infrarouge et de flambage.

### 6.1.4 Plate-forme LAMA

**Participants** : Francis Avnaim, Sabine Moisan, Jean-Paul Rigault, Patrick Itey.

La plate-forme regroupe des outils communs que différents générateurs peuvent partager et que le concepteur peut adapter à la tâche de l'expert. Cette année, les évolutions ont porté principalement sur la bibliothèque BLOCKS de composants réutilisables pour les moteurs et sur l'introduction d'une bibliothèque de composants pour les interfaces graphiques. Le langage YAKL et ses traducteurs ont aussi évolué, pour prendre en compte les évolutions de la bibliothèque et permettre la traduction vers le langage C++.

La bibliothèque BLOCKS fournit au concepteur des primitives de description des comportements d'un moteur de système à base de connaissances (en pilotage pour le moment) à un haut niveau d'abstraction, et permet de représenter de façon uniforme des concepts abstraits, dont le concepteur pourra définir des variantes concrètes, le plus indépendamment possible d'un

langage de programmation. Suite au travail de modélisation réalisé dans le cadre des thèses de M. Marcos [7] et M. Crubézy [6], une analyse de domaine sur les systèmes de pilotage a permis des raffinements dans la partie pilotage de la bibliothèque BLOCKS. Certains composants de la bibliothèque ont, à cette occasion, été reconçus et étendus. Ces composants fournissent une représentation, qui, tout en étant opérationnelle, se situe à un niveau d'abstraction proche du niveau du dialogue entre le concepteur et l'expert. Pour le concepteur, la conformité entre l'implantation et le modèle conceptuel est ainsi facilitée.

De plus, le passage en C++ de la partie pilotage de BLOCKS est terminé, ce qui permet d'avoir une version du moteur PEGASE en C++. L'implantation d'autres moteurs (en commençant par MEDIA) est en cours. Pour cela, nous avons en particulier enrichi les structures de modélisation des plans d'opérateurs générés par un moteur. Un plan peut maintenant être représenté par une structure de graphe permettant de prendre en compte le parallélisme entre opérateurs ainsi qu'un certain nombre d'opérations de manipulation du plan. Les composants développés ont été intégrés à la bibliothèque.

Nos efforts ont aussi porté sur les interfaces graphiques pour les systèmes de pilotage. En effet, le pilotage de programmes nécessite le développement de diverses interfaces conviviales. L'utilisateur du système aura besoin, par exemple, d'avoir une vision de la base de connaissances mais également des plans générés par le moteur et de leur exécution. De même, la création et la mise à jour d'une base de connaissances par les experts du domaine peuvent se faire via une interface graphique.

Une première interface en C++, basée sur IlogViews a été finalisée ; elle a été utilisée pour la version Lisp du moteur PEGASE. À partir de cette interface, nous avons dégagé des composants pour la création d'interfaces graphiques, de manière à offrir une bibliothèque de composants pour les interfaces. Comme pour les moteurs, l'approche utilisée s'apparente à celle des « component frameworks » en génie logiciel. Cette approche permet le développement de plusieurs interfaces à partir des composants de base. De plus, avec le développement du Web, nous envisageons à moyen terme la possibilité d'interroger à distance le système de pilotage dans un schéma client-serveur. Ainsi, n'importe quel utilisateur sur le réseau pourrait utiliser les services d'un serveur dédié abritant l'application. Dans cet optique, le langage Java a été choisi pour le développement des composants de cette bibliothèque. Le développement s'est fait en Java1.2 en utilisant Swing. La bibliothèque pour les moteurs étant développée en C++, la communication entre les langages Java et C++ a utilisé JNI (Java Native Interface) qui est donné en standard dans la JDK.

## 6.2 Interprétation automatique de séquences d'images

**Participants :** Nathanaël Rota, Robert Stahr, Monique Thonnat.

**Résumé :** *Nous nous intéressons ici à l'interprétation dynamique de scènes à partir de séquences d'images, et ce afin d'automatiser la compréhension des activités se déroulant dans une scène donnée. Le capteur utilisé est une caméra fixe et monoculaire, les scènes sont des scènes d'intérieur ou d'extérieur et les objets mobiles sont principalement des êtres humains et des véhicules. L'objectif de nos travaux est, d'une part, la modélisation du processus d'interprétation de séquences*

*d'images et, d'autre part, la validation de ce modèle à travers le développement d'un système générique d'interprétation.*

*Cette année, nous avons travaillé dans le cadre de l'action Dyade Telescope d'une part sur l'enrichissement du nombre et du type d'événements reconnus et d'autre part sur la qualité et la robustesse des traitements d'images en amont de l'interprétation.*

Le problème que nous souhaitons résoudre est de produire une interprétation logique du comportement des personnes dans une scène à partir des images envoyées par une caméra vidéo fixe. Pour cela, nous avons besoin de résoudre deux sous-problèmes.

Le premier est de fournir à chaque image des mesures sur le contenu de la scène. Le système en charge de résoudre ce problème est appelé module perceptuel. Le second sous-problème est de comprendre le contenu, c'est à dire, interpréter les comportements des personnes. Pour cela, on cherche à reconnaître des scénarios prédéterminés à partir de certaines primitives. Le système en charge de résoudre ce second problème est appelé module décisionnel de reconnaissance de scénarios.

Un système d'interprétation vidéo, dont l'architecture abstraite est présenté figure 6, implique trois niveaux de problèmes. Le premier est le choix de modèles adéquats pour résoudre tel ou tel sous-problème. Le deuxième problème est la paramétrisation de ces modèles et le troisième problème réside dans l'organisation globale des différents modèles choisis. Les efforts durant l'année 1999 ont été mis sur ces trois niveaux de la façon suivante. Une librairie d'échange de donnée en XML a été réalisée pour faciliter l'échange entre les différents modèles. Le problème de la paramétrisation des modèles a été abordé sous deux angles par la réalisation d'un logiciel d'aide à la calibration de caméra et par l'étude d'une méthode de paramétrisation dynamique grâce aux techniques de pilotage. L'effort qu'en au choix des modèles a été mis sur les modèles d'événements impliqué dans la reconnaissance de scénario

### 6.2.1 Librairie d'échange de données en format XML

**Participants :** Robert Stahr, Nathanaël Rota.

Un système de vidéo-surveillance intelligente a été développé au sein du projet Orion au cours des cinq dernières années. Plusieurs personnes ont participé au développement de ce système qui commence à avoir une taille assez importante en terme de lignes de code. Nous avons alors commencé à concevoir une restructuration complète du système pour faciliter sa maintenance, pour simplifier l'introduction des nouveaux algorithmes et pour rendre possible l'application des techniques de pilotage de programmes.

L'idée est de séparer la chaîne de traitement actuelle en modules et sous-modules bien indépendants avec des interfaces bien définies. Avec une telle séparation nette, il sera possible pour un module de contrôle extérieur de gérer l'ordre et le choix de modules et les valeurs de leurs paramètres. Ces choix peuvent être guidés par des éventuels modules de vérification qui surveillent les sorties des différents modules. En plus, avec des interfaces claires et bien définies, il sera plus facile de sauvegarder les sorties de chacun des modules sur disque et/ou d'y chercher ses entrées. Avec la possibilité de chercher et de stocker les données intermédiaires sur disque

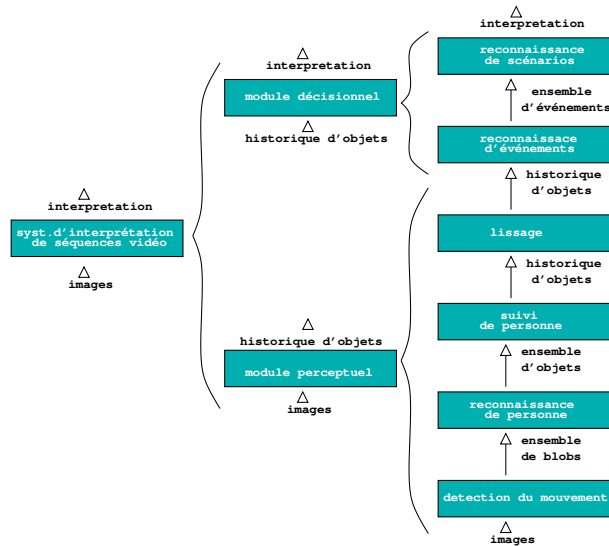


FIG. 6 – Architecture abstraite du système d'interprétation de séquences vidéo. De bas en haut, le flot de données. De gauche à droite, différents niveaux d'abstraction

il sera aussi possible de tester les modules indépendamment les uns les autres et de les tester sur des données synthétiques (des fichiers de données créés à la main).

Pour faciliter la programmation et surtout la maintenance (la stabilité par rapport aux changements) de toutes ces écritures et lectures de fichiers de données et pour faciliter la création des fichiers de données synthétiques, nous avons choisi d'essayer d'utiliser le langage XML.

Nous avons développé (en C) un ensemble de routines pour mettre en correspondance une structure de données C et des données stockées dans un fichier texte sous forme XML. Une fois que la structure centrale de données d'un programme C (avec toutes ses sous-structures en listes chaînées, tableaux ou autres) a été décrite, la description et le nom d'un fichier XML est passé à une routine qui remplit la structure avec les données du fichier. Ceci peut impliquer la création dynamique de toute une hiérarchie de listes chaînées, de tableaux et de structures de différents types. Une routine pour stocker des données déjà présentes en mémoire dans un fichier XML a aussi été développée. Les routines de lecture sont basées sur une librairie nommée « expat » (en C) d'analyse (parsing) de bas niveau des fichiers XML écrite par James Clark et disponible sous licence GNU

### 6.2.2 Pilotage d'une chaîne de traitement d'images

**Participants :** Sébastien Wybo, Nathanaël Rota, Sabine Moisan, Monique Thonnat.

Ce stage avait pour objectif la réalisation d'un programme permettant de piloter la partie de traitement d'images du programme d'interprétation afin de pouvoir améliorer ses résultats. Ce stage s'inscrit ainsi naturellement à l'intersection de deux domaines de recherche : le traitement

de l'image et le pilotage de programmes.

Le pilotage de programmes permet de libérer l'utilisateur du réglage des paramètres des programmes. Or le système actuel utilise de très nombreux paramètres dont les valeurs sont souvent des nombres « magiques » qui correspondent bien à un contexte d'utilisation mais ne seront plus adaptés pour un autre. L'utilisation du pilotage de programmes pourra ainsi permettre d'automatiser la détermination de ces valeurs mais aussi les corriger éventuellement si le contexte venait à changer.

Le langage de description de connaissances YAKL (Yet Another Knowledge Language) qui a été développé dans l'équipe ORION par S.Moisan a été utilisé

Dans ce cadre, deux réalisations ont été faites. D'une part, nous avons bien séparé les programmes de la chaîne de traitement d'images, ce qui a permis un découpage en huit types de programmes, soit un total de quatorze programmes indépendants. D'autre part, une base de connaissances issue de l'expertise de l'équipe a été élaborée. Les tests de traitement d'images effectués par un moteur de pilotage ont montré la faisabilité et la richesse de la méthode.

### 6.2.3 Logiciel d'aide à la calibration de caméra vidéo

**Participants :** Thomas Menguy, Robert Stahr, Nathanaël Rota.

Pour l'interprétation d'images vidéo, il peut sembler important d'extrapoler des informations tridimensionnelles à partir d'images bidimensionnelles ; pour cela, une connaissance précise du passage du 2D au 3D est primordiale.

Formellement, le problème revient à définir une liste de couples de points 2D de l'image et de points 3D de la scène afin d'estimer les paramètres de la caméra, c'est à dire les coefficients de la projection de la scène dans l'image. Cette estimation est effectuée par la méthode des moindres carrés.

Il n'en reste pas moins que l'acquisition d'une liste de couples de points est une étape fastidieuse. Il a donc été réalisé un logiciel graphique d'aide à l'acquisition de ces couples.

Ce logiciel comprend trois parties : l'acquisition de points 2D, l'acquisition de points 3D et la liaison entre points 2D et 3D. L'acquisition de points 2D est effectuée directement dans l'image par de simples clics de souris étiquetés et commentés. L'acquisition de points 3D est faite en ligne, mais sa saisie est facilitée par l'utilisation de plans et de droites prédéfinies. Quant à la liaison entre ces points, elle est réalisée graphiquement par sélection consécutive d'un point 2D et d'un point 3D.

Ce logiciel a été réalisé grâce à une librairie de visualisation et de traitement d'images, développée dans l'équipe, ainsi que la librairie de conception d'interface graphique Xforms : <ftp://einstein.phys.uwm.edu/pub/xforms>

### 6.2.4 Modèle d'événements

**Participants :** Nathanaël Rota, Monique Thonnat.

Le problème final qui nous intéresse consiste à reconnaître de façon incrémentale des scénarios prédéfinis à partir des résultats d'un module perceptuel. Reconnaître un scénario correspond à reconnaître tous les événements qui le composent et vérifier la cohérence des interdé-



pendances. Ces interdépendances peuvent être d'ordre temporel, spatial, logique ou algébrique. Le principe de reconnaissance de scénarios tient en deux temps : on recueille image après image des événements intéressants apparus dans la scène. puis on instancie, avec ces événements, un modèle de scénario. En d'autres termes, la reconnaissance de scénario correspond à la mise à jour d'un ensemble de scénarios partiellement reconnus. Un scénario totalement instancié est déclaré reconnu.

Un événement est une propriété spatio-temporelle du système qui représente les changements significatifs d'état de la scène. Les événements typiques sont «Entrer», «Accélérer», «S'asseoir» ou «Lever un bras». Le principe de l'algorithme de reconnaissance est le suivant : un événement est reconnu si, pour une mesure donnée, appelée état, la valeur de cet état est significativement différente entre une image  $I_0$  et une autre  $I_n$ . L'intervalle de temps entre  $I_0$  et  $I_n$  est appelé délai de reconnaissance  $d_{rec}$ . Le problème de la reconnaissance d'événements se résume donc au problème atemporel de trouver un ensemble d'états décrivant suffisamment précisément la scène.

L'objectif de ce modèle d'état est de fournir un ensemble d'états génériques reposant sur un formalisme qui facilite son extension et sa paramétrisation. Ce modèle d'état est basé sur le formalisme des arbres n-aires. Un état de la scène est un arbre n-aire qui représente la façon dont cet état est calculé. On distingue quatre types de nœuds : les objets, les descripteurs, les opérateurs et les classifieurs. Les feuilles de cet arbre sont nécessairement des nœuds de type objet. Les nœuds terminaux sont nécessairement des de type descripteur, opérateur ou classifieur. Le nœud racine est nécessairement de type classifieur.

- Les objets sont les objets de la scène à un instant  $t$ , c'est à dire des éléments de  $O : \{o_{i,j}\}$ , où  $i$  est la classe de l'objet et  $j$  son étiquette. Par exemple, l'objet  $o_{personne,1}$  est un objet mobile reconnu comme étant une personne et son étiquette est 1,  $o_{equipement,porte}$  est l'objet de la classe équipement étiquetté comme étant la porte. Les trois classes d'objets utilisées sont les personnes, les zones et les équipements.
  - Les personnes sont les objets mobiles de la scène reconnus comme humains. Les traitements préalables nous fournissent un vecteur  $(px_{3D}, py_{3D})$  représentant la localisation de la personne sur le sol de la scène, un vecteur  $(vx_{3D}, vy_{3D})$  représentant le vecteur vitesse de la personne ainsi qu'une mesure de sa taille  $h$ .
  - Une zone est un objet statique représentant une partie du sol de la scène sous forme d'un polygone  $\{(px_i, py_i) \mid i = 1 \dots k\}$ . Les zone sont des objets obtenus par modélisation externe de la scène (voir figure 8).
  - Un équipement représente tous types d'objets volumiques de l'environnement dont nous connaissons la base sous forme d'un polygone  $\{(px_i, py_i) \mid i = 1 \dots k\}$  et la hauteur  $h$ . À l'instar des zones, les équipements sont obtenus par modélisation externe de la scène (voir figure 8).
- Les descripteurs sont des fonctions définies de  $O$  dans  $R^p$  dont le but est de pouvoir accéder à une mesure faite sur un objet. Par exemple, la taille, la position, la vitesse, la forme, la trajectoire, l'orientation, le volume ou le poids sont des descripteurs possibles. Cette notion assure l'ancrage du modèle dans les résultats numériques du module perceptuel. Nous avons utilisé 4 nœuds de type descripteur : *pos*, *taille*, *vitesse*, *forme*:

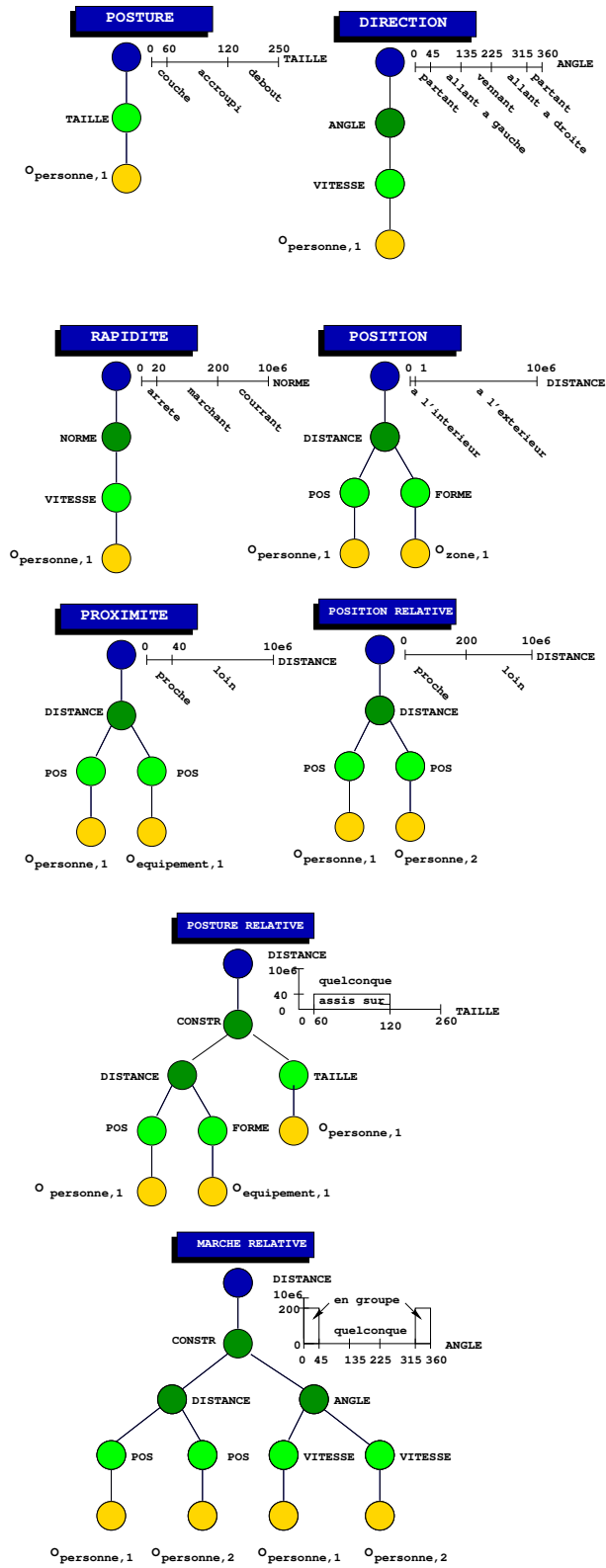


FIG. 7 – Huit instances du modèle d'état. En jaune les objets, en vert clair les descripteurs, en vert foncé les opérateurs et en bleu les classifieurs

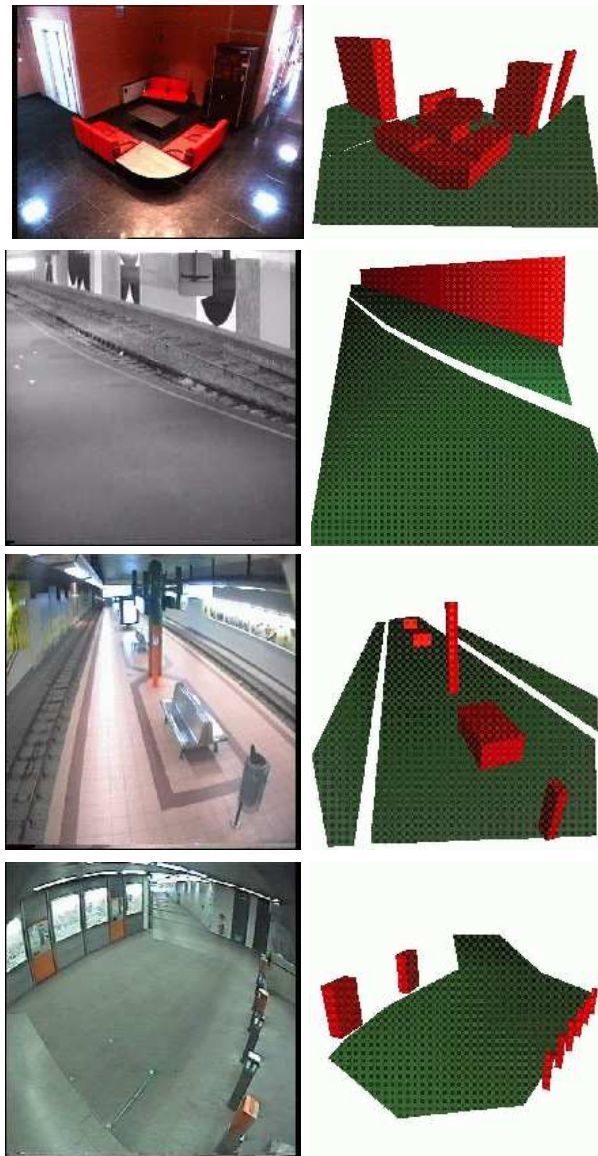


FIG. 8 – À gauche: les images des scènes vides, à droite: la modélisation 3D des scènes. De haut en bas: un « coin café », un quai de métro simple, un quai de métro central et le hall d'une station de métro.

- Les opérateurs sont des fonctions définies de  $(R^{p_1} \times \dots \times R^{p_n})$  dans  $(R^{q_1} \times \dots \times R^{q_n})$  permettant de faire des opérations sur les mesures. On peut citer, à titre d'exemple, des opérateurs tels que les mesures de distance, de normalisation, les opérateurs arithmétiques classiques ou les fonctions logiques. Nous avons utilisé 4 nœuds de type opérateur : *distance, norme, angle, constr*
- Les classifieurs sont des fonctions définies de  $(R^p)$  dans  $S$ , l'ensemble des symboles :  $\{grand, petit, rapide, lent, proche, loin, etc.\dots\}$ . Ces opérateurs assurent le passage du numérique vers le symbolique en associant à chaque valeur symbolique un domaine de définition numérique. Ce domaine de définition représente les paramètres de l'état. Nous avons utilisé 8 nœuds de type classifieur : *posture, direction, rapidite, position, proximite, position relative, posture relative, marche relative*.

Étant donné ce modèle, on instancie à chaque image un ensemble d'états génériques prédéfinis avec les objets possibles existant à cet instant dans la scène. Il en résulte un ensemble d'états instanciés qui fournissent une description de la scène à cet instant. On repasse au problème de la reconnaissance d'événements en comparant ce nouvel ensemble avec ceux obtenus aux instants précédents. Les états ayant changé de valeur symbolique donnent lieu à la création d'un événement.

À partir de ces opérateurs et de ces objets, nous avons défini 8 exemples d'états : *posture, direction, rapidite, position, proximite, position relative, posture relative* et *marche relative* (voir figure 7) et 18 événements : *chute, s'accroupit, se leve, part a droite, part a gauche, s'en va, arrive, s'arrete, se met a marcher, se met a courrir, sort de, entre dans, s'approche de, s'eloigne de, s'assoit sur* et *evoluent en groupe*. Nous avons testé la robustesse de ces 18 événements issus de 8 états sur différentes séquences d'images de longueurs variables entre 100 et 400 images, de tailles variables et dans quatre environnements différents. Il semble que l'utilisation d'une logique d'intervalles stricts pour la modélisation des classifieurs soit source d'erreurs. En effet lorsque la valeur numérique d'un état fluctue autour de la frontière d'un domaine de définition d'une valeur symbolique, on observe des sauts entre deux valeurs symboliques et donc la génération de faux événements. Une logique d'intervalles flous pourrait minimiser ce phénomène. En second lieu, l'hypothèse de  $d_{rec} = 1$  restreint les événements de l'instance que nous avons testée à être reconnus entre deux images consécutives. Cette hypothèse semble trop restrictive pour des événements mettant en jeu des états dont on connaît *a priori* l'inertie. Les événements produits à partir de  $direction(o_{personne,i})$  ou de  $rapidite(o_{personne,i})$  seraient plus intéressants s'ils étaient calculés sur un intervalle de 1 seconde par exemple. À part ces deux points, les principaux problèmes rencontrés sont imputables aux erreurs des traitements antérieurs. Plus un état dépend de résultats des modules de début de chaîne, plus il est robuste. Plus un état dépend de résultats de milieu de chaîne, moins il est robuste. On trouve quatre catégories d'états : les états ne dépendant que de la qualité de la localisation, les états dépendant de la localisation et du mode de lissage, les états dépendant de la localisation et de la reconnaissance d'objets et les états dépendant de la localisation, de la reconnaissance et du suivi.

### 6.3 Reconnaissance semi-automatique de pollens

**Participants :** Alain Boucher, Nicolas Chleq, Monique Thonnat.

Le projet ASTHMA a démarré en octobre 1998 pour une durée de trois ans. Ce projet a pour but de prévoir et de fournir des informations sur la qualité de l'air pour les personnes sensibles aux aéroallergènes (pollens et autres poussières présents dans l'air). Dans le cadre de ce projet, le projet Orion s'intègre dans l'étude d'un système de reconnaissance semi-automatique de pollens, et ce en collaboration avec le LASMEA de Clermont-Ferrand et le réseau espagnol d'aérobiologie (REA), représenté par des équipes de Cordoue et de Barcelone.

Le but du système étudié est d'automatiser la tâche de lecture et de comptage des grains de pollens au microscope, qui est longue et répétitive pour les palynologues (spécialistes des pollens). Le projet Orion s'intéresse au développement d'algorithmes de reconnaissance 3D de pollens à l'aide d'une base de connaissances sur les différents taxons et leurs caractéristiques.

En effet, une première étude effectuée en 1997 avait conclu sur la nécessité de travailler en 3D [Maz97]. L'observation des méthodes de travail des palynologues aboutit aux mêmes conclusions car l'information nécessaire pour la reconnaissance et la classification d'un grain de pollen peut se situer à différents niveaux à l'intérieur du grain. Ce grain peut également être vu sous différents angles et aspects, selon son orientation lorsqu'il se dépose sur le capteur. Les images que nous possédons sont en fait une série de coupes optiques du grain de pollens, ces coupes étant espacées de 0.5 à 1 micron pour un grain d'une taille variant de 15 à 60 microns.

Il est également nécessaire de posséder une base de connaissances sur les différents taxons étudiés pour arriver à les reconnaître. Cette base, développée conjointement avec les experts en palynologie, contient les différentes données morphologiques et géométriques pour les pollens étudiés. Elle contient également des informations sur les sites de captage de ces pollens, c'est-à-dire sur les taxons les plus susceptibles d'être rencontrés en fonction de la période de l'année.

Le travail de développement des algorithmes de l'INRIA se décompose en trois phases :

- Le *pré-traitement des images 3D* pour extraire les caractéristiques importantes des pollens pour la reconnaissance. Les primitives détectées sont de niveaux intermédiaires, puisque des connaissances spécifiques sont déjà utilisées. Par exemple, nous tentons d'extraire les pores et les sillons présents sur différentes coupes. Certaines méthodes développées sont directement inspirées du travail d'analyse des palynologues. La figure 9 montre quelques résultats préliminaires pour cette phase.
- La *construction d'un modèle 3D du grain de pollen* sur lequel nous inscrivons toutes les caractéristiques reconnues. Ce modèle permettra de raisonner spatialement sur la nature du grain pour en permettre la classification. Ce modèle permet de pallier les informations incomplètes ou erronées qui ont pu être extraites des images dans la première phase de traitement. Des hypothèses de construction de ce modèle sont envisagées par le système, qui raffine son modèle au fur et à mesure du traitement.
- La *classification automatique des grains de pollen*, se basant sur le modèle 3D précédemment construit et sur les informations contenues dans la base de connaissances. Si le système ne reconnaît pas un grain de pollen, il est prévu de l'annoter comme tel et de permettre à un technicien de l'identifier a posteriori (mode semi-automatique).

À ce stade du projet, seule la première phase a produit des résultats pratiques. La deuxième phase en est à l'étude théorique et la troisième phase n'a pas encore commencé.

---

[Maz97] M. MAZIÈRE, « Étude de faisabilité pour la reconnaissance automatique de grains de pollen », 1997, Rapport de DEA, université de Nice-Sophia Antipolis.

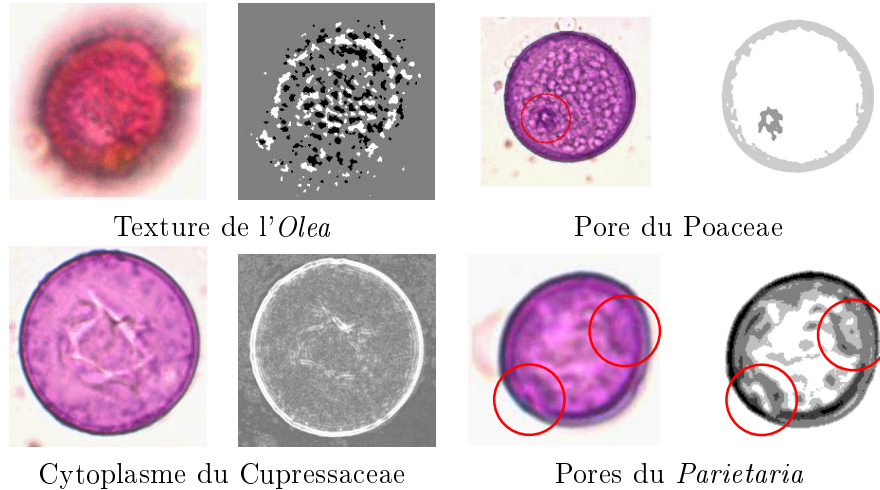


FIG. 9 – Résultats préliminaires pour l'extraction de caractéristiques essentielles pour la reconnaissance et la classification de grains de pollen. Ces caractéristiques sont extraites sur différentes coupes optiques du pollen, d'où la nécessité d'effectuer le travail de reconnaissance en 3D.

### 6.3.1 Analyse de la forme de grains de pollen à partir de plusieurs plans images

**Participants :** Nicolas Dey, Alain Boucher, Monique Thonnat.

Pour mieux comprendre la formation de l'image afin de l'exploiter ultérieurement dans un but de reconnaissance, nous avons **modélisé la formation de l'image d'un pollen**, qui est un objet microscopique **translucide** et tridimensionnel. Du fait de la translucidité du spécimen, les zones situées au-dessus et en-dessous de la zone focalisée (qui apparaît nette) contribuent aussi à la formation de l'image, mais en dégradant la qualité de l'image avec un apport de «flou». Pour reproduire le phénomène de la formation de l'image d'un pollen, il est nécessaire de modéliser non seulement l'**objet** et le **microscope** mais aussi le **flou** avant de construire l'**image**.

Le pollen a été modélisé comme étant une succession discrète de **plan-objets**  $o(x,y,z)$  accolés les uns aux autres selon la direction de l'axe optique du système ( $z$ ) : ils présentent tous une distribution en **intensité** bidimensionnelle ( $x, y$ ) correspondant au coefficient d'absorption réel du pollen.

C'est le microscope qui est responsable de la perte de résolution selon l'axe  $z$ , car il va avoir pour effet de former une image 2-D de la réalité 3-D. La Réponse Impulsionnelle Incohérente (R.I.I.) du microscope est une constante que l'on va noter  $h_M(x,y)$ . Mais il est évident que tous les plans-objets ne peuvent pas être focalisés (nets) en même temps : le flou doit intervenir à travers une R.I.I. notée  $h_F(x,y,\epsilon)$  où  $\epsilon$  est la **défocalisation**. On a choisi un modèle gaussien (symétrique) déjà existant.

L'image a été modélisée comme étant une **somme discrète** « **d'imagettes** », une imagette  $i(x,y,\epsilon)$  étant donnée par la convolution de  $o(x,y,z)$  avec les différentes R.I.I.. Des résultats de cette modélisation pour différentes focalisation sont présentés dans la figure 10.

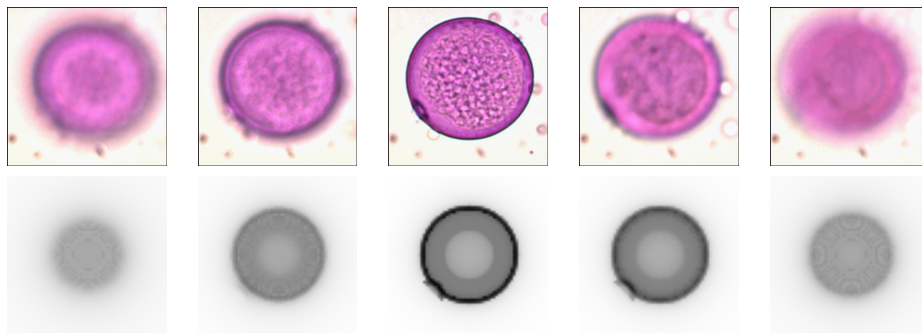


FIG. 10 – *Comparaison de coupes optiques réelles (en haut) avec la modélisation de la formation de l'image (en bas) d'un grain de pollen.*

## 7 Contrats industriels (nationaux, européens et internationaux)

**Participants :** Nicolas Chleq, Monique Thonnat.

**Résumé :** *En 1999, Orion a continué à participer au projet européen ASTHMA et à l'action Dyade Telescope. Un nouveau projet européen en vidéosurveillance ADVISOR de 3 ans a été monté en cours d'année et démarrera en janvier 2000.*

- Coopération de 2 ans avec Bull dans le cadre de l'action de développement Dyade Telescope pour le développement d'un système intelligent de vidéosurveillance.
- Projet ASTHMA : ce projet d'une durée de 3 ans et d'un montant de 1.2 MF pour Orion a pour objectif de faciliter la détection d'aéroallergènes.

Pour plus d'informations sur le projet européen ASTHMA, voir aussi la rubrique concernant les actions financées par la Commission Européenne.

## 8 Actions régionales, nationales et internationales

### 8.1 Actions régionales

- M. Thonnat est membre du conseil d'administration de l'UFR STAPS de la faculté des sciences du sport de l'université de Nice-Sophia Antipolis.

### 8.2 Actions nationales

- Participation à l'action coopérative LiRe regroupant les projets Syco et Omega de Nancy, le projet Sherpa de Grenoble et le projet Orion de Sophia Antipolis sur le thème de la résolution de problèmes avec limitation de ressources. Un article commun Sherpa-Orion sur ce thème a été accepté à la conférence RFIA 2000.

- Participation à l'action coopérative Mediaspace regroupant l'équipe PRIMA de Jim Crowley de l'ENSIMAG à l'INRIA Monbonnot, l'équipe de Joelle Coutaz à l'Université Joseph

Fourrier à Grenoble et le projet Orion sur le thème de l'interprétation de séquences d'images appliquée à des environnements de travail collectif distants ou mediaspaces.

- Coopération étroite avec l'équipe du professeur R. Di Paola de l'INSERM -Unité 494 La Pitié-Salpêtrière, Paris pour le pilotage de programmes de traitement de séquences d'images médicales. Collaborateurs principaux : Florent Aubry, Virginie Chameroy.

- Coopération avec l'équipe d'Alain Almar du Cesta au CEA de Bordeaux sur le thème du pilotage de programmes appliqué à des codes de calcul scientifique. Cette coopération est effectuée dans le cadre du co-encadrement de la thèse de Jean-Christophe Noël par A. Almar et M. Thonnat.

- Collaboration avec le CEA Saclay (DIST/SIIA), dans le cadre du groupe de travail « Connaissances et codes ».

- Collaboration avec le pôle Cindyniques de l'ENSMP Sophia Antipolis, dans le cadre du co-encadrement de la thèse de Sabine Delaitre par S. Moisan et J-L. Wybo. L'objectif de la thèse est de recueillir et d'utiliser l'expérience pour réviser des plans d'actions dans des situations de danger, avec une application sur les feux de forêts.

### 8.3 Actions financées par la Commission Européenne

**Résumé :** *Le projet a participé cette année à un projet européen de type Climat et Environnement en interprétation d'images : le projet ASTHMA.*

#### 8.3.1 Projet ASTHMA

Ce projet de type Climat et Environnement a pour objectif de fournir aux utilisateurs des informations précises et quasi temps-réel sur les aéroallergènes (dont les pollens) et la qualité de l'air afin de les aider à optimiser leurs traitements thérapeutiques. Les partenaires sont le laboratoire pharmaceutique Zambon (Italie), ACRI (France), les universités de Barcelone (Espagne), de Cordoue (Espagne), de Clermont (France), le CHU de Nice (France), le FISBAT (Italie) et le PAMOC (France). La contribution d'Orion concerne la reconnaissance automatique d'images 3D de pollens observées sous microscope optique.

### 8.4 Relations bilatérales internationales

#### 8.4.1 Europe

- Nous coopérons avec l'équipe du professeur del Pobil, responsable du groupe IA et Robotique de l'université de Jaume I, à Castellón (Espagne) sous la forme du co-encadrement par S. Moisan et A. del Pobil de la thèse de M. Marcos sur le thème de la validation et de la vérification de systèmes à base de connaissances pour le pilotage de programmes.

#### 8.4.2 Afrique

Nous coopérons avec Wided Chaari et Faouzi Ghorbel du Grift à l'ENSI de Tunis dans le cadre d'un projet STIC franco-tunisien de deux ans sur le thème des systèmes à base de connaissances distribués pour la vision.



## 9 Diffusion de résultats

### 9.1 Animation de la communauté scientifique

- M. Thonnat a été relecteur des revues Artificial Intelligence, IJCV (International Journal on Computer Vision), CVIU (Journal of Computer Vision and Image Understanding), Pattern Recognition Letters et de la revue RIA (Revue d'Intelligence Artificielle).

- S. Moisan a été relecteur de la revue Artificial Intelligence.

- M. Thonnat fait partie du comité d'organisation de la conférence internationale FG2000 Face and Gesture recognition.

- S. Moisan fait partie du comité d'organisation de la conférence ECOOP'2000 sur la programmation par objets (responsable des workshops).

- S. Moisan fait partie du comité de programme de la conférence IC'2000, sur l'ingénierie des connaissances.

-M. Thonnat a participé aux jurys de thèse de Jérôme Gout au LAAS à Toulouse comme rapporteur, de François Bérard à l'université Joseph Fourier à Grenoble comme rapporteur, de Régis Tomczak à l'université Blaise Pascal comme rapporteur, de Sivia Codareshi à l'université de Linköping en Suède comme opposant et de Corine Cossart-Jaupitre de Supaero à Toulouse comme examinateur.

-S. Moisan a participé au jury de thèse de Sueli Ferreira à l'université Joseph Fourier à Grenoble, comme rapporteur et de Philippe Haik à l'université Henri Poincaré, Nancy I, comme rapporteur.

### 9.2 Enseignement universitaire

- DEA Informatique et ESSI à l'université de Nice-Sophia Antipolis, cours d'Intelligence artificielle (6h S. Moisan). Orion est équipe d'accueil pour le DEA Informatique de l'UNSA.

- École Supérieure des Sciences de l'Ingénieur, Sophia Antipolis, cours Conception orientée objet (18h S. Moisan), cours métaprogrammation (6h S. Moisan).

### 9.3 Thèses en cours

- Nicolas Dey :Analyse 3D d'images de pollens pour la prévention des allergies, université de Nice-Sophia Antipolis.

- Monica Crubézy : Pilotage de programmes pour le traitement de séquences d'images médicales, université de Nice-Sophia Antipolis. Thèse soutenue en février 1999.

- Mar Marcos : Vérification et validation de systèmes à base de connaissances en pilotage de programmes, université de Castellón, Espagne. Thèse soutenue en février 1999.

- Jean-Christophe Noël : Pilotage de programmes pour les codes de simulation en calcul scientifique, université de Nice-Sophia Antipolis. Thèse soutenue en novembre 1999.

- Nathanaël Rota : Interprétation de séquences video pour la reconnaissance de gestes et de scénarios, université de Nice-Sophia Antipolis.

## 9.4 Participation à des colloques, séminaires, invitations

N. Rota et R. Stahr ont participé à la conférence CVPR, et au workshop en vidéosurveillance Colorado (USA).

M. Thonnat a été invitée du 27 au 28 mars au colloque TAIMA99 à Hammamet (Tunisie) sur les techniques d'analyse d'images.

M. Thonnat a été invitée par le Prof Matsuyama du 17 au 24 novembre pour rencontrer les équipes de recherches de Kyoto et d'Osaka et présenter une conférence invitée au 3<sup>ème</sup> International Workshop on Cooperative Distributed Vision les 19 et 20 novembre à Kyoto (Japon).

M. Thonnat a été invitée par l'Université de Linköping (Suède) du 13 au 16 décembre pour présenter un séminaire sur l'interprétation de séquence d'images et être opposant de la thèse de S. Codareshi.

## 10 Bibliographie

### Ouvrages et articles de référence de l'équipe

- [1] V. CLÉMENT, M. THONNAT, « A Knowledge-Based Approach to Integration of Image Procedures Processing », *CVGIP: Image Understanding* 57, 2, mars 1993, p. 166–184.
- [2] S. LIU, P. SAINT-MARC, M. THONNAT, M. BERTHOD, « Feasibility study of automatic identification of planktonic foraminifera by computer vision », *Journal of foraminiferal research* 26, 2, avril 1996, p. 113–123.
- [3] M. THONNAT, V. CLÉMENT, J. OSSOLA, « Automatic Galaxy classification », *Astrophysical Letters and Communication* 31, 1-6, 1995, p. 65–72.
- [4] M. THONNAT, M. GANDELIN, « Un système expert pour la description et le classement automatique de zooplanctons à partir d'images monoculaires », *Traitement du signal, spécial I.A* 9, 5, novembre 1992, p. 373–387.
- [5] M. THONNAT, S. MOISAN, « Knowledge-based systems for program supervision. », in: *First international workshop on Knowledge-Based systems for the (re)Use of Programs libraries KBUP'95*, INRIA, p. 4–8, Sophia Antipolis, France, mars 1995.

### Thèses et habilitations à diriger des recherches

- [6] M. CRUBÉZY, *Pilotage de programmes pour le traitement d'images médicales*, thèse de doctorat, université de Nice-Sophia Antipolis, février 1999.
- [7] M. MARCOS, *Verification and Validation of Program Supervision Systems*, thèse de doctorat, université de Jaume I (Castellón, Espagne), février 1999.

### Articles et chapitres de livre

- [8] M. MARCOS, S. MOISAN, A. P. DEL POBIL, « Knowledge Modeling of Program Supervision Task and its Application to Knowledge Base Verification », *Applied Intelligence* 10, 2/3, March/April 1999, p. 181–192.
- [9] C. SHEKHAR, S. MOISAN, R. VINCENT, P. BURLINA, R. CHELLAPPA, « Knowledge-based Control of Vision Systems », *Image and Vision Computing* 17, 8, mai 1999, p. 667–683.

---

**Communications à des congrès, colloques, etc.**

- [10] M. THONNAT, S. MOISAN, M. CRUBÉZY, « Experience in Integrating Image Processing Programs », *in: International Conference on Computer Vision Systems*, Las Palmas, Canary Islands, janvier 1999.
- [11] M. THONNAT, N. ROTA, « Image Understanding for Visual Surveillance Applications », *in: Third International Workshop on Cooperative Distributed Vision*, Kyoto, Japan, novembre 1999.

**Divers**

- [12] N. DEY, « Analyse de la forme de grains de pollen à partir de plusieurs plans image », 1999, Rapport de DEA Signaux, Images et Communications de Toulouse.
- [13] T. MENGUY, « Calibre: interface de saisie pour la calibration de caméras », 1999, Rapport de stage, Supelec.
- [14] S. WYBO, « Pilotage d'une chaîne de traitement d'image », 1999, Rapport de stage ENS Cachan.