

# Un système multi-agent pour l'analyse d'images

Yazid Abchiche et Patrice Dalle

IRIT

(Institut de Recherche en Informatique de Toulouse),

Equipe TCI, Université Paul Sabatier,

118, Route de Narbonne,

31062, Toulouse Cedex 4

France.

Télé+ 33 (0)5 61 55 63 20, Fax+ 33 (0)5 61 55 62 58

Yazid.Abchiche@irit.fr, Patrice.Dalle@irit.fr

**Résumé** Cette recherche se situe dans le cadre de la conception interactive et incrémentale d'applications de traitement et d'analyse d'images. Cet article décrit un système d'analyse d'images (SAI) basé sur une architecture multi-agent<sup>1</sup> où chaque opérateur d'analyse d'images est encapsulé par un agent. Pour résoudre un problème posé par l'utilisateur, le système fait évoluer sa configuration de manière à constituer un graphe d'opérateurs capable d'extraire l'information demandée dans l'image. Cette évolution est réalisée de manière incrémentale via des cycles d'interaction et d'exécutions au cours desquels l'utilisateur précisera son objectif en fonction des résultats présentés par le système.

**Mots clés** Analyse d'images, configuration, agent, système multi-agent.

---

<sup>1</sup> Cette recherche est menée en collaboration avec l'équipe SMAC (Systèmes Multi-Agents Coopératifs) de l'IRIT, dirigée par Pierre Glize.

## 1 Introduction

Nous présentons un système d'analyse d'images basé sur une architecture multi-agent. Cette recherche se situe dans le cadre de la conception interactive et incrémentale d'applications de traitement et d'analyse d'images.

Notre objectif étant de construire des applications spécialisées et adaptatives, plutôt que d'étudier un système général d'interprétation [7,8], nous nous intéressons à l'étude d'un environnement de développement. Dans ce cadre, trois problèmes doivent être abordés : la spécification et la formulation du problème à résoudre et de l'objectif à atteindre [2], la configuration d'un graphe d'opérateurs (planification) [3, 5] et le pilotage de l'exécution de ce graphe et de son adaptation au contexte [9]. Le plus souvent, les recherches se focalisent sur un seul de ces problèmes, parfois deux. Notre approche vise à prendre en compte ces trois problèmes; en particulier, elle intègre explicitement les interactions avec le concepteur d'application. Elle s'appuie sur deux éléments : un formalisme de représentation des concepts construits par les opérateurs de TI [6] et une architecture multi-agents pour implémenter ces opérateurs [1]. Le formalisme facilite l'articulation entre spécification et configuration : d'une part, il permet de construire des outils d'interaction exploitables par l'utilisateur et d'interpréter ces interactions; d'autre part il permet aux agents d'évaluer leur contribution à l'objectif final. Les apports d'une architecture multi-agents sont bien connus [4] (facilité de construction et de mise à jour, possibilité de modéliser le parallélisme, capacité de focalisation, résolution de problèmes hétérogènes et fiabilité). Dans notre approche elle fait émerger la configuration adéquate du comportement coopératif des agents [11] et elle permet de conserver un comportement adaptatif lors de l'exécution.

## 2 Principe de l'approche

Pour résoudre un problème posé par l'utilisateur (par exemple identifier et localiser un objet), le système fait évoluer sa configuration de manière à constituer un graphe d'opérateurs capable d'extraire l'information demandée dans l'image. Cette évolution est réalisée de manière incrémentale via des cycles d'interaction et d'exécutions au cours desquels l'utilisateur précisera son objectif en fonction des résultats présentés par le système. Le système de base est réalisé en encapsulant chaque opérateur par un agent dont le comportement est essentiellement coopératif. La configuration du graphe d'opérateurs résulte donc de l'auto-organisation du collectif d'agents. Les agents transforment l'information et attachent aux entités produites une description formelle de l'effet de ces transformations. Un système secondaire exploite cette description pour piloter le dialogue avec l'utilisateur et pour interpréter et traduire ses requêtes. La compétence des agents d'analyse d'images est représentée via les descripteurs de propriétés qu'ils peuvent manipuler. Leur évolution résulte des interactions entre agents et des réactions de l'utilisateur aux résultats obtenus, transmis par le système secondaire. Après une présentation générale du système, nous décrirons plus en détail le système multi-agent d'analyse d'images.

## 3 Présentation du système

Nous allons décrire brièvement le système général de conception d'applications. Il comporte trois modules principaux l'*interface utilisateur*, le *système secondaire* et le *système multi-agent d'analyse d'image* qui est l'objet de cet article et dont nous décrirons ensuite le fonctionnement plus en détail.

### 3.1 Interface avec l'utilisateur

Cette interface gère le dialogue avec l'utilisateur en usant de plusieurs formes d'interactions adaptées. Elle permet à l'utilisateur de parler des données (entités □ pixels, région, amas de régions, etc.) et de leurs contenus [10]. Elle exploite le langage de description des concepts [6] pour les présenter et les manipuler à l'aide d'indices visuels (formes, orientation, aspect, etc.). Elle permet de manipuler les données de façon interactive et de définir des concepts incrémentalement. Cette interface fait l'objet d'une recherche spécifique.

### 3.2 Système secondaire

Il joue le rôle de médiateur entre l'interface et le système d'analyse d'image Il remplit une double fonction de représentation et d'anticipation :

- Il décrit l'évolution du système d'analyse d'images.
- Il détecte des stabilités représentant des concepts de TI. Il attache du sens aux entités créées par le SAI en les décrivant comme des instances de concepts du domaine d'application.
- Il transmet les entités associées aux concepts au système d'interface.
- Il interprète les réactions de l'utilisateur à la présentation des concepts en cours d'élaboration.
- Il traduit ces réactions en termes de description de concepts à obtenir et de nouvelle organisation du système pour les obtenir. Cette description va influencer l'évolution du SAI vers l'élaboration de concepts correspondant mieux aux notions du domaine que le concepteur cherche à décrire.

L'interface et le système secondaire font encore l'objet de recherches. Leur fonctionnement s'appuie sur les résultats fournis par le SAI que nous allons décrire.

### 3.3 Système d'analyse d'images

Son rôle est d'enchaîner, de combiner et de contrôler des opérateurs. Chaque opérateur remplit une fonction de traitement d'image, de structuration de données ou de construction d'entités à différents niveaux de complexité. Ce système est réalisé sous la forme d'un système multi-agent. Avant d'être spécialisé, il présente *a priori* une forte combinatoire. Ses caractéristiques sont les suivantes □

- Granularité faible : l'unité de base est la fonction de traitement d'image, qui peut être très élémentaire (c'est-à-dire qu'elle met en jeu un seul descripteur).
- Configurable (possibilité de construire des macro-opérateurs).
- Fortement connecté.
- Comportement autonome : le système cherche à réaliser une fonction globale de structuration des données et d'agrégation des fonctions.

## 4 Modélisation des agents du système d'analyse d'image

Chaque opérateur est encapsulé dans un agent. Ce dernier est composé d'un ensemble de modules qui lui permettent d'interagir avec son environnement.

- (a) Base de connaissances □ elle représente la croyance en soi (connaissances relatives à l'opérateur encapsulé par l'agent, ses entrées, ses sorties) et elle décrit les services rendus par l'agent et son expertise dans le domaine, c'est à dire tout ce qui lui permet d'agir pour répondre à une requête d'un autre agent. Une valeur est attribuée initialement à chaque possibilité, que l'agent augmente ou diminue en fonction de ses interactions avec son environnement. Elle évolue avec son expérience dans le temps □
- La phase d'apprentissage permet d'ajuster les paramètres liés à l'opérateur de TI et de donner une configuration type pour le domaine d'expertise.
  - L'interaction avec l'expert du domaine permet de préciser la configuration type afin de faire émerger des concepts plus précis du domaine.

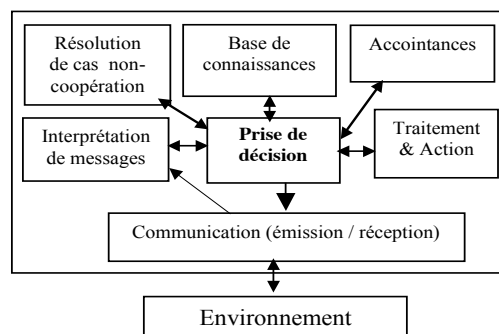


Figure 1 : Modèle d'agent du système d'analyse d'images.

- (b) Communication □ émission et réception des messages entre l'agent et son environnement. Il contient les procédures d'envoi et de réception de messages.
- (c) Interprétation des messages □ vérification de l'absence d'ambiguïtés dans le message et préparation à l'exploitation.
- (d) Prise de décision □ noyau de l'agent. Ce module prend une décision en fonction de ce qui lui sera retourné par le module d'interprétation de messages. (Ex : lancer le traitement en cas de non ambiguïté et de possibilité de satisfaction de la requête de l'émetteur ou reporter le traitement dans le cas contraire). Il met à jour les croyances : la tâche de conception consiste à faire évoluer, pour chaque agent, sa base de connaissance (ajustement des paramètres de l'opérateur associé) et les liens d'échanges avec les autres agents pour obtenir une organisation satisfaisante c'est-à-dire la configuration d'un graphe d'opérateurs dont l'exécution produit un résultat satisfaisant l'objectif. Cette évolution se traduit par la variation du degré de croyance en fonction de l'orientation donnée par le système médiateur et d'interactions internes, entre agents. Ce module sélectionne les agents fournisseurs : l'agent fait appel à tous les agents qui peuvent produire la donnée dont il a besoin en entrée, sélectionne une des données proposées, d'après ses croyances. Enfin, ce module met à jour les accountances (en entrée et en sortie) dans le cas de création d'un nouvel agent (ou de changement des croyances) en fonction des messages qui lui seront transmis par le module de d'interprétation de messages suite à son interaction avec les autres agents.
- (e) Résolution des cas de non coopération □ résolution des problèmes de conflit, d'inutilité et de concurrence décrits au §5.
- (f) Module de traitement □ mécanismes de lancement et de contrôle d'exécution de l'opérateur.
- (g) Accountances □ l'agent contient une liste des agents en relation directe avec lui. Chaque accountance est composée de six champs □
1. <Id> □ identifiant qui permet à l'agent d'interagir directement avec cette accountance,

2. *<Inputs>* □ paramètres de cette accointances (agent) en entrée. Ils fournissent la description de la donnée en entrée et des paramètres de l'opérateur encapsulé.
3. *<Outputs>* □ paramètres de cette accointances (agent) en sortie. Ils fournissent la description de la donnée en sortie.
4. *<Belief>* : degré de croyance attribué à cette accointances; il varie suivant les différentes interactions,
5. *<Type>* □ type de l'agent (cognitif, réactif) qui définit le mode d'interaction avec cette accointance,
6. *<Services>* □ services rendus par cette accointances. Ils décrivent la transformation liée à l'opérateur encapsulé.

## 5 Modélisation du système d'analyse d'images

Les opérateurs d'analyse d'images sont implémentés sous forme d'agents coopératifs. La bibliothèque d'opérateurs est une société d'agents qui cherchent à se configurer pour répondre à l'objectif donné. Le principe de la collaboration entre les agents est la coopération : la force des accointances est modifiée suivant la possibilité d'exploitation des résultats de l'action d'un agent par les agents récepteurs (principe d'utilité). Pour cela, chaque agent cherche à détecter les situations non coopératives, qui sont [11] :

- Le conflit □ accès de plusieurs agents à une ressource commune (modification d'une image commune), production de résultats contradictoires.
- La concurrence □ situation où deux ou plusieurs agents répondent à la même requête.
- L'inutilité □ production d'une donnée qui finalement n'est pas utilisée par les autres agents.

La tâche de conception consiste à faire évoluer les liens d'échanges entre ces agents vers une organisation «optimale» c'est-à-dire qui répond au problème posé et telle qu'il n'existe pas d'autre organisation de graphe plus simple qui produise le même résultat. Cette évolution est le résultat d'interactions avec l'utilisateur qui réagit aux résultats présentés, en fonction de son objectif, et d'interactions internes, entre agents, générées à partir d'une modélisation de l'analyse d'images.

## 6 Fonctionnement

L'architecture du système est à base d'agents autonomes travaillant d'une manière concurrente (*multi-thread*). Chaque agent de ce système doit être capable de mettre en œuvre un modèle générique de raisonnement s'appuyant sur des comportements de localisation, focalisation, reconnaissance et interaction avec les autres agents.

Nous avons testé ce système en mode autonome, c'est-à-dire dans lequel un but complet est fourni au système, celui-ci évoluant jusqu'à la construction d'entité correspondant au but. Pour l'initialiser, des agents (en général les agents d'identification) reçoivent une requête qu'ils propagent dans tout le système (en fonction de leurs accointances), en la reformulant, jusqu'à ce que d'autres agents aient assez d'information pour déclencher un traitement (en général les agents qui reçoivent en entrée l'image à traiter). Ceux-ci peuvent alors effectuer leur action et fournissent des résultats qu'ils propagent en retour. Les données se répandent dans tout le réseau, jusqu'à ce que les agents qui ont reçu la requête initiale aient les informations suffisantes pour y répondre de manière satisfaisante.

En fonction des évaluations réalisées par les agents d'interprétation, une information d'utilité est propagée en retour; elle permet de modifier les poids des liens entre agents de manière à ce qu'à terme, l'opérateur le plus pertinent soit exécuté, avec le paramétrage adapté.

Pour éviter un comportement combinatoire, le système doit recevoir de son environnement des réactions dont la prise en compte va orienter la configuration des liens de la société d'agents ainsi que le paramétrage des opérateurs. Deux éléments interviennent dans cet environnement :

- la structure des entités de TI représentée par un formalisme de description syntaxique des transformations réalisée par les opérateurs; elle contraint les relations et les actions des agents-opérateurs de TI,
- l'interprétation par l'utilisateur des données produites par le TI, et les contraintes imposées par cet utilisateur pour que ne soient validées que les entités interprétables.

## 7 Conclusion et perspectives

La conception d'une application spécialisée consiste à configurer le graphe des opérateurs adéquats. Cette configuration résulte de l'auto-organisation d'un système composé d'agents ayant essentiellement une attitude sociale coopérative. Ce système s'organise et se stabilise en fonction des réactions provenant de son environnement. Nous avons réalisé une maquette et évalué le système sur un jeu d'opérateurs. Nous avons simulé les retours de l'utilisateur par des artefacts. Le système se configure en exploitant toutes ses possibilités de résolution. Dans la version complète, en cours de développement, les interactions système - utilisateur sont prises en charge par un ensemble système médiateur. D'autre part, pour augmenter la compétence des agents, nous complétons leur capacité de traitement (déclenchement des opérations de TI) par une capacité de manipulation formelle des descriptions symboliques des entités produites par ces traitements.

## Références

- [1] Y. Abchiche, Conception d'une plate-forme pour la configuration d'un graphe d'opérateurs de traitement d'images par système multi-agent, *DEA Informatique de l'Image et du Langage, UPS, Toulouse, France*, juin 1999.
- [2] O. Capdevielle, P. Dalle, Image processing chain construction by interactive goal specification, *First IEEE ICIP, Austin, Texas USA*, pp.816-820, November 1994.
- [3] R. Clouard, A. Elmoataz, C. Porquet, M. Revenu, Borg: a knowledge-based system for automatic generation of image processing programs, *IEEE Trans on PAMI*, vol 21, n° 2, pp.128-144, February, 1999.
- [4] D. Crevier, R. Lepage, Knowledge-Based Image Understanding System: A Survey, *Proceeding of Vision and Image Understanding*, Vol. 67, n° 2, pp.161-185, Academic Press, August 1997.
- [5] P. Dalle, P. Dejean, Planification en Traitement d'Image : Approche basée sur les données, *11ème RFIA Clermont-Ferrand, France*, tome II pp. 75-84, janvier 1998.
- [6] P. Dejean, P. Dalle, Image analysis operators as concept constructors, *SwSIAT'96, San Antonio, Texas USA*, pages 66-70, April 1996.
- [7] C. Dillon et T. Caelli , Learning image annotation : the CITE system *Videre : Journal of computer vision research*, MIT Press, vol 1, n° 2, 1998.
- [8] J-M. Jolion, Les systèmes de Vision, *Hermès* 2001.
- [9] S. Moisan, R. Vincent., M. Thonnat : Program supervision : from knowledge modeling to dedicated engines, Rapport INRIA n° 3324, 1997.

- [10] A. Nouvel, P. Dalle, Description des entités du traitement d'images pour la conception interactive d'applications, ORASIS 2001, pp. 349-358, Cahors, France, juin 2001.
- [11] SMAC, Une théorie de la coopération pour l'auto-organisation des systèmes artificiels, IRIT-UPS, Rapport n° 97-58-R, décembre 1997.