



**HAL**  
open science

# Intégration d'un système de vision dans un îlot auto-controlé de fabrication manufacturière

Bruno Rochotte

► **To cite this version:**

Bruno Rochotte. Intégration d'un système de vision dans un îlot auto-controlé de fabrication manufacturière. Informatique [cs]. Université Henri Poincaré - Nancy 1, 1992. Français. NNT : 1992NAN10033 . tel-01753467

**HAL Id: tel-01753467**

**<https://hal.univ-lorraine.fr/tel-01753467v1>**

Submitted on 29 Mar 2018

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

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



## AVERTISSEMENT

Ce document est le fruit d'un long travail approuvé par le jury de soutenance et mis à disposition de l'ensemble de la communauté universitaire élargie.

Il est soumis à la propriété intellectuelle de l'auteur. Ceci implique une obligation de citation et de référencement lors de l'utilisation de ce document.

D'autre part, toute contrefaçon, plagiat, reproduction illicite encourt une poursuite pénale.

Contact : [ddoc-theses-contact@univ-lorraine.fr](mailto:ddoc-theses-contact@univ-lorraine.fr)

## LIENS

Code de la Propriété Intellectuelle. articles L 122. 4

Code de la Propriété Intellectuelle. articles L 335.2- L 335.10

[http://www.cfcopies.com/V2/leg/leg\\_droi.php](http://www.cfcopies.com/V2/leg/leg_droi.php)

<http://www.culture.gouv.fr/culture/infos-pratiques/droits/protection.htm>

U.F.R. STMIA

Groupe de Formation Doctorale : Automatique, Electronique, Electrotechnique

Thèse présentée  
à l'université de Nancy I  
pour l'obtention du grade de  
Docteur de l'Université de Nancy I  
Option "Production Automatisée"  
par  
Bruno Rochotte



**Intégration d'un système de vision  
dans un Ilot Auto-Contrôlé  
de Fabrication Manufacturière**

soutenue publiquement le 12 février 1992 devant la commission d'examen composée de :

Président	M. VERON	Professeur à l'Université de Nancy I
Directeur de thèse		Directeur du CRAN
Rapporteurs	R. HUSSON	Professeur à l'INPL
	P.L. WENDEL	Professeur à l'ENSPS
Examineurs	J. BREMONT	Professeur à l'Université de Nancy I
	J. CICOTELLI	Chef de service au CTBA, Pont à Mousson
	B. DUFAY	Directeur Délégué, ITMI
	G. MOREL	Maître de conférences à l'Université de Nancy I, directeur de recherche associé

A AMARIA  
A GOLIATH  
A MERCURE

Le travail de recherche présenté dans ce mémoire a été réalisé au Centre de Recherche en Automatique de Nancy (CRAN - URA - 821 du CNRS) au sein de l'équipe "Automatisation Intégrée de Production" du Laboratoire d'Automatique et de Commande Numérique (LACN) dans le cadre d'un Contrat de recherche MRT n° 88 p 0718.

Je souhaite exprimer mes sincères remerciements aux personnes qui me font l'honneur de composer le jury :

Monsieur le Professeur VERON, Directeur du CRAN et du LACN, pour m'avoir accueilli ces dernières années au sein du laboratoire et pour la confiance qu'il me fait en acceptant de présider ce jury,

Monsieur le Professeur R. HUSSON, de l'INPL Nancy, et Monsieur le Professeur P.L. WENDEL, de l'ENSPS Strasbourg, pour l'intérêt qu'il ont manifesté à l'égard de ce travail dont ils ont daigné être les Rapporteurs,

Monsieur le Professeur J. BREMONT, de l'Université de Nancy I, Monsieur J. CICOTELLI, Docteur Ingénieur et Chef de service au CTBA et Monsieur B. DUFAY, Directeur Délégué ITMI, pour la bienveillance qu'ils ont manifesté en examinant ces travaux,

Monsieur G. MOREL, Maître de Conférences à l'Université de Nancy I et Directeur de Recherche associé, sans qui ce manuscrit ne saurait être ce qu'il est.

Je tiens enfin à exprimer ma profonde sympathie à tous les membres du laboratoire qui ont participé, par leur soutien moral, à la réalisation de ce travail.

**RESUME** : ce mémoire de recherche fait une proposition d'architecture physique d'Ilot Auto-Contrôlée de Fabrication Manufacturière intégrant un système de vision **multi-utilisateurs, multi-fonctions, intégré, ouvert et auto-contrôlé**. Sur cette base physique, nous proposons à la fois des **Méthodologies de Conception** et des **Modèles de Référence** appliqués aux points de vue **informationnel et fonctionnel** de la visionique considérée comme un domaine intégré dans le Système de Production. L'ensemble des modèles proposé (informationnels et fonctionnels) contribue à la formalisation du **savoir-faire** (sémantique) métier de la visionique, utile dans une optique de réutilisation (reconception).

**MOTS CLES** : Visionique, CIM, Capteur Intelligent, Méthodologie, Modèle de Référence Fonctionnel, Modèle de Référence Informationnel, Design Management and Manufacturing System, Ingénierie

**ABSTRACT** : this research work proposes a Loop-Controlled Flexible Manufacturing Cell Architecture integrating a **multi-users, multi-functions, integrated, open and loop-controlled** vision system. Through this physical basis, we propose **Design Methods and Reference Models** according to the **Informational and Functional** vision views in which vision system is considered as a production sub-system. The Informational and Functional Reference Models must allow to safeguard the vision field's "**know-how**", needful in a twice use (reusability).

**KEYWORDS** : Vision, CIM, Intelligent Sensor, Method, Functional Reference Model, Informational Reference Model, Design Management and Manufacturing System, Engineering

# Sommaire

## Introduction

Contexte de l'étude.....	1
Emergence du Génie visionique .....	2
Notre démarche.....	3

## Chapitre I : Concepts d'intégration

1. Problématique .....	1
2. Concepts d'intégration organisationnelle .....	2
2.5. Les Modèles de Référence .....	2
2.6. Modèle de Référence de Système Intégré de Production .....	4
2.7. Concepts d'intégration horizontale .....	6
2.7.3. Control, Maintenance and technical Management System (C.M.M.S.).....	6
2.7.4. Design Management and Manufacturing Système (D.M.M.S.)..	8
2.8. Prototypage de validation du projet MRT .....	8
3. Les concepts d'intégration méthodologique .....	9
3.1. Le projet IMPACS [CHE 90].....	9
3.2. Le cadre CIM OSA [GAC 90].....	11
3.3. Application au Génie Visionique .....	12
3.3.1. "Vue fonctionnelle" du "domaine vision".....	12
3.3.2. "Vue informationnelle" du "domaine vision" .....	13
3.3.3. Situation de nos travaux dans le modèle CIM-OSA .....	13
4. Les outils et méthodes utilisés .....	14
4.1. Le formalisme Entité/Relation Etendu, l'outil MEGA .....	14
4.2. Le formalisme SADT, l'outil ORCHIS .....	18
4.3. Le formalisme GRILLE, l'outil ORCHIS-GRILLE .....	18
4.4. Les formalismes et l'outil SPEX.....	22

## Chapitre II : Vision et informations techniques

1. Problématique .....	1
2. Les éléments de l'acquisition d'image.....	2
2.1. La source .....	4
2.2. L'objet .....	5
2.3. Le récepteur.....	7
3. Les éléments de l'interprétation d'image.....	10
3.1. Analyse de différents systèmes de vision.....	11
3.1.1. Intermediate Symbolic Representation ISR [BRO 89].....	12

3.1.2. Knowledge-Based Computer Vision [GOO 89].....	13
3.1.3. Thin Line Code (TLC) [ JAG 89] .....	13
3.1.4. Powervision [CON 89].....	14
3.1.5. Base de données pour la représentation d'objets plats [QIN 88].....	15
3.1.6. Base de données image pour une interprétation sémantique [PIZ 89].....	16
3.1.7. Conclusion sur l'analyse de différents systèmes de vision .....	16
3.2. Traitements et données .....	17
3.2.1. La numérisation et les prétraitements.....	20
3.2.2. La segmentation et l'interprétation.....	20
3.2.3. La représentation des traitements.....	22
3.2.4. La représentation du monde visuel.....	23
3.2.5. Les programmes et les données utilisés par les traitements .....	25
3.2.6. Les données issues du traitement .....	26
4. L'individu SYSTEME DE VISION .....	28
5. Application : système de vision matricielle .....	28
5.1. Les optiques.....	29
5.2. Les capteurs.....	29
5.3. Les éclairages .....	29
5.4. Les scènes.....	29
5.5. Les traitements.....	31
5.5.1. Les prétraitements .....	31
5.5.2. La segmentation .....	32
5.5.3. L'interprétation .....	32
5.6. Conception du Système d'Information intégrant le contrôle qualité par vision.....	34
5.6.1. MCD de Référence .....	34
5.6.2. MCD partiel .....	34
5.6.3. MOD particulier .....	36
5.6.4. MLD particulier.....	37
5.6.5. Génération du code SQL.....	38

### **Chapitre III : Intégration dans le système physique de production**

1. Problématique .....	1
2. La cellule flexible .....	2
2.1. modèle de référence des processus opératoires.....	2
2.2. Le niveau cellule.....	3
2.3. Le niveau station.....	5
3. Le rôle d'un système de vision dans la cellule.....	5
3.1. La vision comme poste de contrôle de la pièce dans la cellule .....	6
3.2. La vision comme outil de contrôle de la pièce et des processus opératoires sur une machine.....	6
3.3. La vision "réflexe" .....	7
4. Objectifs pour la vision .....	7
4.1. Un capteur flexible, multi-fonctions, multi-utilisateurs .....	7
4.2. Un capteur intégré.....	8
4.2.1. Aspect communication .....	8
4.2.2. Aspect contrôle commande.....	8
4.3. Un capteur auto-contrôlé.....	8

4.4. Un capteur ouvert .....	9
4.5. En bref : les yeux de la cellule .....	10
5. Décomposition fonctionnelle.....	11
5.1. Fonction préparer triplet "source-objet-recepteur".....	11
5.2. Fonction capter information .....	11
5.3. Fonction traiter l'information .....	13
5.4. Fonction contrôler l'opération de vision .....	13
5.5. Les données nécessaires à l'exécution d'une opération de vision.....	13
6. Les objets "intelligents" en automatique.....	14
6.1. Le concept de filtre .....	15
6.2. Le module fonctionnel d'automatisme.....	16
6.3. L'actionneur "intelligent" .....	17
6.4. Le capteur "intelligent" .....	18
7. Application à l'intégration physique d'un système de vision dans une cellule flexible d'usinage .....	18
7.1. L'architecture physique de la cellule .....	18
7.1.1. L'ordinateur cellule.....	19
7.1.2. La station du CHARLY-ROBOT .....	19
7.1.3. La station POSTE DE BRUT.....	21
7.1.4. Les équipements partagés .....	21
7.2. Les contrôles réalisés .....	21
7.2.1. Les contrôles sur le POSTE DE BRUT .....	21
7.2.2. Les contrôles sur le poste d'usinage CHARLY-ROBOT.....	22
7.3. Exemple : contrôle d'un perçage sur fraiseuse CHARLY-ROBOT .....	22
7.3.1. Fonction "actionner lampe" .....	26
7.3.1.1. Prototypage .....	26
7.3.1.2. Réalisation .....	26
7.3.2. Fonction "positionner caméra".....	28
7.3.3. Fonction "prendre image + prétraiter + transférer" .....	29
7.3.3.1. Prototypage .....	29
7.3.3.2. Réalisation .....	29
7.3.4. Fonction "segmenter + interpréter image".....	30
7.3.5. Fonction "contrôler l'opération de vision" .....	31
7.3.5.1. Les traitements de contrôle de l'opération de vision..	31
7.3.5.2. Les données de contrôle de l'opération de vision .....	32
7.3.5.2.1. MCD de Référence .....	32
7.3.5.2.2. MCD partiel.....	32
7.3.5.2.3. MOD particulier.....	32
7.3.5.2.4. MLD particulier et code SQL.....	34

#### **Chapitre IV : Intégration dans le système de gestion de production**

1. Problématique .....	1
2. Concevoir les produits.....	2
3. Concevoir les processus.....	3
3.1. L'approche par variante (dérivation).....	4
3.2. L'approche générative .....	4
3.3. L'approche Généralisation/Particularisation .....	4
3.4. Situation des méthodes de conception des processus dans le "trièdre méthodologique".....	5
3.5. Deux exemples de système de conception des processus .....	5

3.5.1. Le système expert PROPEL.....	5
3.5.2. Le projet Quick Turnaround Cell manufacturing - QTC.....	6
3.5.2.1. Modélisation du produit (design).....	6
3.5.2.2. Planification des processus (Process planning) [KAN 89].....	7
3.5.2.3. Inspection par vision (vision inspection) [PAR 88a,b].....	7
3.6. Proposition d'une démarche de conception d'un MCD pour la conception des processus intégrant le procédé de vision .....	8
3.6.1. L'individu SPECIFS VISION.....	12
3.6.2. Les individus FORME et MATIERE .....	12
3.6.3. Les individus CONTROLE, PRECISION, CADENCE .....	13
3.6.4. L'individu ENVIRONNEMENT .....	14
4. Concevoir les moyens .....	14
4.1. Systèmes de conception des "connaissances" consommées par un système de vision.....	15
4.1.1. Quick turnaroud cell manufacturing - QTC [PAR 88a][PAR 88b].....	15
4.1.1.1. Modélisation et interface .....	15
4.1.1.2. Préparation du traitement .....	16
4.1.2. CADG-Based Computer Vision [HAN 87].....	17
4.1.2.1. La représentation géométrique .....	17
4.1.2.2. La sélection automatique des paramètres de reconnaissance .....	17
4.1.2.3. La synthèse de l'arbre de stratégie .....	18
4.1.3. Conception automatisée de système de vision [KAT 87] [KAT 88].....	19
4.1.3.1. Modélisation d'une attitude de pièce.....	19
4.1.3.2. Modélisation des senseurs (capteurs).....	20
4.1.4. 3D Part Orientation System [BOL 86] .....	20
4.1.4.1. Modélisation de l'objet .....	20
4.1.4.2. Sélection des primitives.....	21
4.1.5. Knowledge-Based-Robotics [KAK 87] .....	22
4.1.5.1. Représentation spéciale capteur (sensor-tuned- representation) .....	22
4.1.5.2. Segmentation d'image 3D et comparaison .....	23
4.1.6. Conclusion sur les systèmes de conception des "connaissances" consommées par un système de vision et proposition d'un MCD .....	23
4.2. Un cycle de conception d'un processus de contrôle par vision .....	26
4.2.1. Le besoin en génie visionique .....	26
4.2.1.1. La notion de cycle de vie.....	27
4.2.1.2. La réutilisabilité .....	28
4.2.1.3. Les objets .....	28
4.2.2. Application à la visionique .....	28
4.2.2.1. La phase de spécification .....	29
4.2.2.2. La phase d'analyse.....	29
4.2.2.3. La phase de conception .....	30
4.2.2.4. La phase de codage .....	30
5. Notre approche pour le prototypage du contrôle-commande de niveau station intégrant le procédé de vision.....	30

5.1. Prototypage du contrôle par vision des perçages de la pièce NASA sur le CHARLY-ROBOT .....	31
5.1.1. Génération de la gamme d'usinage par PROPEL.....	31
5.1.2. Insertion des opérations de contrôle dans la sous-phase .....	33
5.1.3. Prototypage sur SPEX.....	35
5.1.3.1. Programmation Fonctionnelle.....	35
5.1.3.2. Fusion des actinomies .....	37
5.1.3.3. Utilisation sur SPEX .....	40
5.1.3.3.1. L'application .....	40
5.1.3.3.1.1. Le niveau équipement (environnement).....	40
5.1.3.3.1.2. le niveau station (système).....	41
5.1.3.3.2. Prototypage du contrôle-commande de la station.....	41
5.1.3.3.2.1. Démarche .....	41
5.1.3.3.2.2. Boîte Fonctionnelle "CONTROLE PAR VISION" .....	43
5.1.3.3.2.3. Boîte Fonctionnelle "USINAGE" .....	43
5.1.3.3.2.4. Boîte Fonctionnelle des autres procédés.....	43
5.1.3.3.2.5. Fusion des Boîtes Fonctionnelles.....	46
5.1.3.3.2.6. Boîte Fonctionnelle "SOUS-PHASE" .....	46
5.2. Conclusion sur la méthode de prototypage .....	47

**Conclusion**

**Bibliographie**

**Liste des figures**

**Abréviations**

# Introduction

Contexte de l'étude.....	1
Emergence du Génie Visionique .....	2
Notre démarche.....	3

# Introduction

## 1. Contexte de l'étude

Le projet pilote d'**Îlot Automatisé de Production de Pièces Mécaniques** [ADI 83] a été un des supports essentiels de recherches menées par le CRAN/LACN. Les étapes de ce projet [MOR 90]:

- îlot automatisé
- îlot robotisé
- îlot flexible (Flexible Manufacturing System)
- îlot intégré, ouvert (Computer Integrated Manufacturing)
- îlot auto-contrôlé (Manufacturing Intelligence)

montrent l'évolution "expansionniste" des concepts en productique. Parallèlement est apparu "l'îlot méthodologique". En effet l'ensemble des travaux précédents a permis de faire le constat d'une carence en méthodes et outils d'ingénierie d'intégration. De ce fait, notre laboratoire a porté son effort de recherche sur ce thème : le **Génie Automatique**.

*"Le Génie automatique est l'ensemble des questions scientifiques et techniques posées par l'application concrète des théories de l'automatique en milieu industriel"* [FRA 87].

Le Génie Automatique est à l'automatique ce que le Génie Logiciel fut à l'informatique [MEY 86]. Dans cette optique nous préférons cette définition :

*"On appelle Génie Automatique l'application de méthodes scientifiques au développement de théories, méthodes, techniques, langages et outils favorisant la production de systèmes automatisés de qualité"* [MOR 88].

La création d'outils méthodologiques pour la conception des Machines et Systèmes Automatisés de Production pose alors le problème de leur intégration

au sein d'une structure plus large : l'entreprise. La conception des systèmes CIM (Computer Integrated Manufacturing) engendre de nouveaux concepts d'intégration tels que les "architectures et modèle de référence de l'entreprise" [ROB 88]. Les efforts de recherche se sont à la fois portés sur la conception de ces modèles de référence et sur le développement de méthodologies supportant l'ensemble de la démarche d'intégration.

## 2. Emergence du Génie Visionique

Parallèlement, dans les laboratoires de recherche en visionique, les travaux s'orientent vers des applications de plus en plus "pointues" : architecture des matériels de traitement [WEN 87], couplage robot-vision, robotique-mobile [ELA 88], interprétation de scènes naturelles, poursuite de cible [HOR 89], détection de contour [LEV 89], etc... Ces recherches, indispensables en vision, ne sont pas toujours transférables de façon "massive" vers l'industrie comme le furent, par exemple, les machines à commande numérique. En effet, la vision par ordinateur est encore aux yeux des industriels un domaine complexe caractérisé par son aspect "high tech"; ce qui est en partie justifié.

Intégrer une application de vision dans un environnement industriel signifie se heurter à quantité de problèmes :

- l'insuffisance de logiciels (problème de conception des applications)
- l'insuffisance d'échange d'informations avec d'autres systèmes
- la formation du personnel, la pluralité des métiers concernés (optique, informatique, éclairage, automatique)
- les problèmes de maintenance.

A l'évolution technologique des recherches en vision s'associe l'attitude "réunioniste" du Génie Productique qui tente d'intégrer les différents **Génies X**. La position privilégiée de notre laboratoire nous a permis d'envisager un transfert de l'automatique vers la visionique. Ce mariage méthodologique entre "vision" et "productique" a donné naissance à un nouveau thème de recherche : le **Génie Visionique** [VOG 91] dont la signification exacte dans notre laboratoire serait plutôt : **Génie Automatique appliqué à la Visionique**. Nous ne cherchons pas à

améliorer le procédé de vision en tant que tel : nous cherchons à appliquer à la vision les outils et méthodes de la productique afin d'obtenir une "vision intégrable".

### 3. Notre démarche

Nous avons donc décomposé notre étude en quatre chapitres :

- *chapitre I : Concepts d'intégration*, où nous posons les bases de notre démarche à travers un exposé sur les concepts d'intégration. Ces concepts nous amènent à considérer l'intégration du système de vision dans le Système Intégré de Production de l'entreprise à travers deux domaines : le système physique et le système de gestion, selon deux points de vue : fonctionnel et informationnel. Nous présentons ensuite les outils et méthodes utilisés dans la suite du mémoire;

- *chapitres II : Vision et informations industrielles*, où il est question de l'étude des systèmes vision à la fois à travers leur système physique (optique, éclairage, caméra) mais aussi à travers leurs traitements associés afin de modéliser l'ensemble des informations techniques propres au domaine de la vision. L'ensemble de ces informations techniques servent de base à l'intégration du système de vision dans le Système d'Informations Industrielles du Système Intégré de Production;

- *chapitre III : Intégration dans le système physique de production*, où il est question de l'étude du système de vision intégré dans un environnement de cellule flexible. Nous utilisons une méthode de conception d'un système de vision basée sur l'emploi d'outils et méthodes issus du Génie Automatique pour aboutir à une proposition d'architecture physique et fonctionnelle d'un système de vision intégré dans une cellule flexible d'usinage respectant les concepts exposés au chapitre I;

- *chapitre IV : Intégration dans le système de gestion de production*, où nous nous sommes limités à l'étude des activités de conception du système de gestion. Ce chapitre fait le point sur l'état de l'art des méthodes de conception des processus et des moyens appliquées aux systèmes de vision. Parallèlement nous faisons une proposition de modélisation des informations spécifiques à ces

activités. Enfin nous apportons une contribution au Génie Visionique en proposant une méthode de prototypage du contrôle-commande d'un système de vision intégré dans une cellule flexible;

## Chapitre 1

# Concepts d'intégration

1. Problématique .....	1
2. Concepts d'intégration organisationnelle .....	2
2.5. Les Modèles de Référence .....	2
2.6. Modèle de Référence de Système Intégré de Production .....	4
2.7. Concepts d'intégration horizontale .....	6
2.7.3. Control, Maintenance and technical Managment System (C.M.M.S.) .....	6
2.7.4. Design Management and Manufacturing Système (D.M.M.S.)..	8
2.8. Prototype de validation du projet MRT .....	8
3. Les concepts d'intégration méthodologique .....	9
3.1. Le projet IMPACS [CHE 90] .....	9
3.2. Le cadre CIM OSA [GAC 90] .....	11
3.3. Application au Génie Visionique .....	12
3.3.1. "Vue fonctionnelle" du "domaine vision" .....	12
3.3.2. "Vue informationnelle" du "domaine vision" .....	13
3.3.3. Situation de nos travaux dans le modèle CIM-OSA .....	13
4. Les outils et méthodes utilisés .....	14
4.1. Le formalisme Entité/Relation Etendu, l'outil MEGA .....	14
4.2. Le formalisme SADT, l'outil ORCHIS .....	18
4.3. Le formalisme GRILLE, l'outil ORCHIS-GRILLE .....	18
4.4. Les formalismes et l'outil SPEX .....	22

## Chapitre 1

# Concepts d'intégration

### 1. Problématique

L'intégration d'un système de vision dans un environnement manufacturier demeure jusqu'à présent un problème "épineux" et difficile à définir en raison de la vaste étendue des activités qu'il concerne (conception, fabrication, système d'informations, planification, etc ...). Par quel "bout" doit-on envisager l'intégration de la vision? Par où commencer? Nous sommes obligés de définir clairement et de manière formelle le cadre de notre étude.

La productique, ayant été confrontée au même problème, possède déjà ses propres solutions que nous voulons réutiliser : le concept de CIM (Computer Integrated Manufacturing)

*"On entend par CIM une démarche globale en entreprise pour améliorer la productivité des outils industriels. Cette démarche est appliquée à l'ensemble des activités depuis la conception jusqu'à la livraison et l'après-vente dans une optique intégrée, mettant en oeuvre méthodes et moyens (techniques informatiques et automatiques) pour assurer simultanément :*

- l'amélioration de la productivité,
- la diminution des coûts,
- le respect des délais,
- la qualité du produit,
- la flexibilité de tout ou partie d'un ensemble industriel, mobilisant et impliquant l'ensemble des acteurs." [DOU 90].

Ce concept de CIM amène à considérer l'emploi d'une méthodologie impliquant l'utilisation :

- de **formalismes** de modélisation et d'**outils** graphiques associés,
- de **modèles** et **architectures** de référence,
- d'une **approche structurée** (méthodologie).

Par extension, le CIM devient le CIME (Engineering) avec la création de nouveaux outils et méthodes d'ingénierie appliqués au CIM.

Nous présentons donc dans ce chapitre les principales recherches menées actuellement en productique :

- les **concepts d'intégration organisationnelle** où l'on propose des modèles de référence servant à modéliser l'entreprise,
- les **concepts d'intégration méthodologique** où l'on propose des méthodologies de conception des systèmes CIM.

Nous cadrerons nos travaux dans ce contexte.

Nous faisons ensuite une présentation des outils et méthodes utilisés dans l'ensemble du mémoire.

## **2. Concepts d'intégration organisationnelle**

Afin de faciliter la communication des différentes fonctions de l'entreprise, les années 70-80 ont vu se développer des architectures de référence basées sur une décomposition hiérarchique des diverses fonctions de l'entreprise [DOU 91].

### **2.5. Les Modèles de Référence**

*"la définition d'une architecture de référence fournit aux concepteurs des systèmes CIM des modèles génériques de l'entreprise. L'analyse et la conception d'un Système Automatisé de Production, par instanciation de ces modèles, est ainsi cohérente et plus rapide"* [GAC 90];

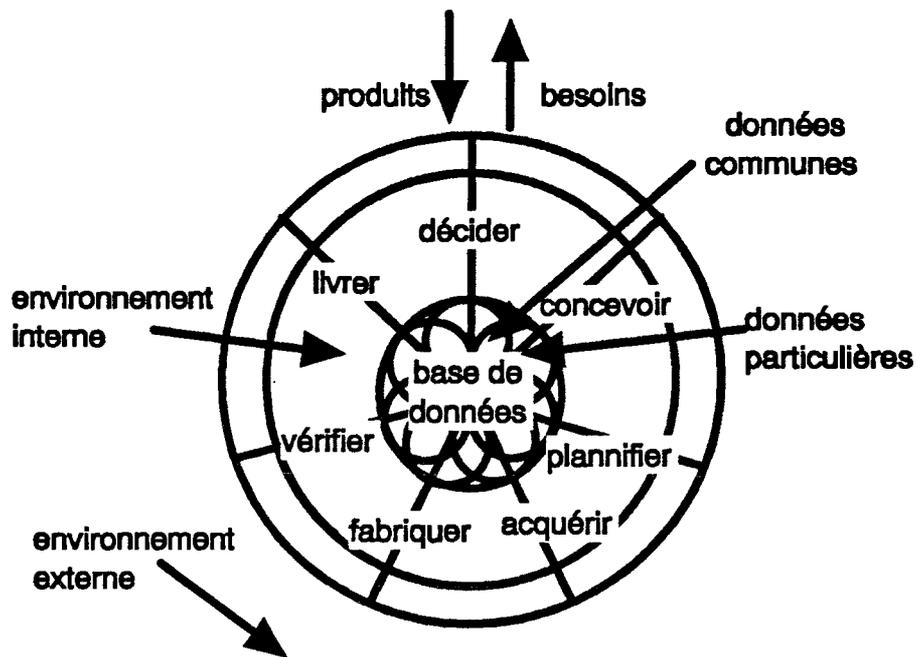
De nombreux travaux [DOU 91] concernant ces modèles ont été effectués :

- le modèle ICAM (Integrated Computer Aided Manufacturing) développé aux Etats-Unis de 78 à 83 dont l'objectif était de présenter les principales

fonctions de production ainsi que les relations entre elles. L'approche est hiérarchique et utilise le formalisme IDEF0 (comparable à SADT) pour le modèle fonctionnel, IDEF1 pour le modèle de données et IDEF2 pour le modèle de simulation,

- le modèle CAMI (cf. figure 1.1.) propose une vision globale de l'entreprise en la décomposant en quatre niveaux (Entreprise, Usine, Atelier, Centre de coût) où sept activités (Décider, Concevoir, Planifier, Acquérir, Faire, Vérifier, Livrer) se partagent une base de données. Ce modèle ne possède pas de formalisme associé et est difficilement implantable,

- le modèle NIST (National Institute of Standards and Technology) (cf. figure 1.2.) propose une décomposition de l'usine en cinq niveaux hiérarchiques : Usine, Atelier, Cellule, Station de travail et Machine. Ce modèle est couramment employé pour représenter les systèmes physiques. Il n'intègre pas les aspects décisionnels.



*figure 1.1. Modèle CAMI*

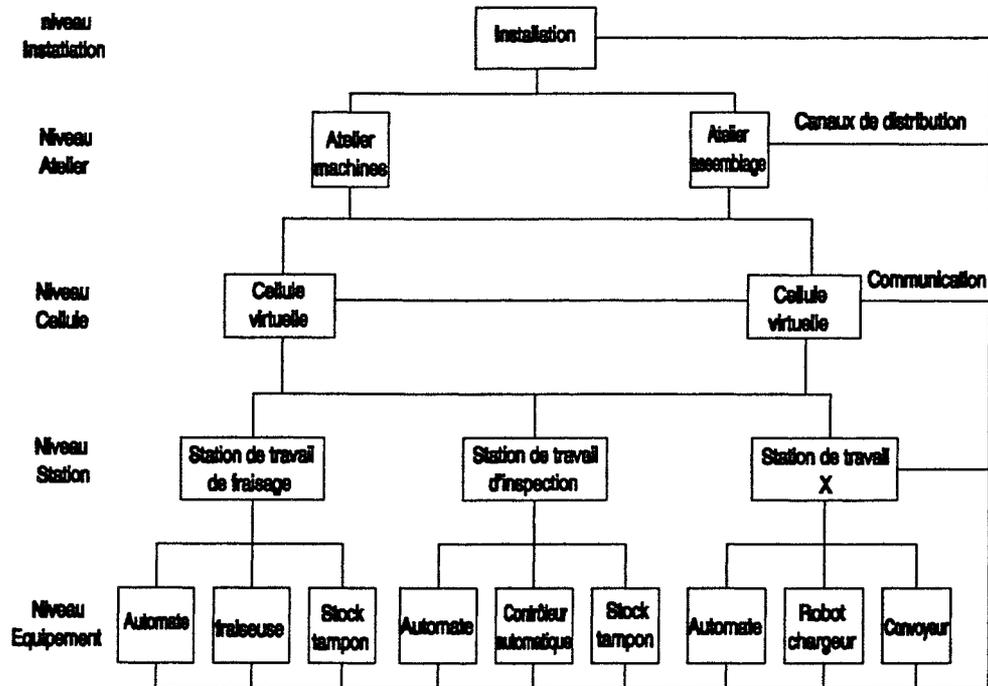


figure 1.2. Modèle NIST

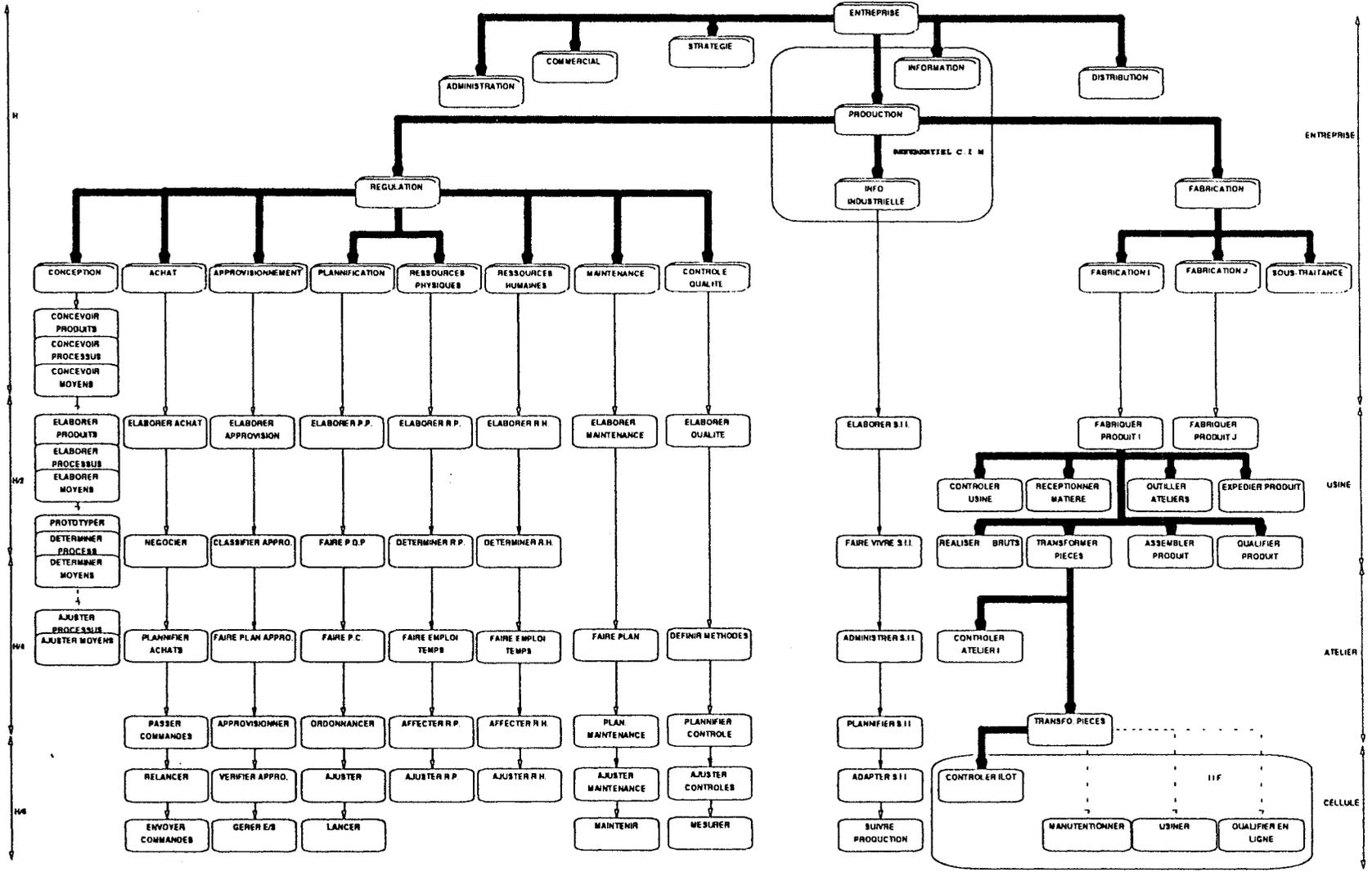
## 2.6. Modèle de Référence de Système Intégré de Production

Dans le cadre de notre étude, nous utilisons le Modèle Conceptuel de Référence Informationnel de l'Entreprise défini par [MOR 89] à partir de [ROB 88] (cf. figure 1.3.). Ce modèle décompose le Système de Production de l'Entreprise en trois domaines :

- le système de régulation de la production
- le système physique de production,
- le système d'informations industrielles.

Il fait apparaître les deux axes d'intégration verticale et horizontale du S.I.P.. L'intégration horizontale décompose le S.I.P en domaines : conception, approvisionnement, planification, maintenance, contrôle qualité.... L'intégration verticale coordonne et décompose chaque domaine en intervenants.

Figure 1.3. Modèle Conceptuel de Référence Informationnel de l'Entreprise [MOR 89]



## **2.7. Concepts d'intégration horizontale**

### **2.7.3. Control, Maintenance and technical Management System (C.M.M.S.)**

Le concept de CMMS, développé dans le cadre du projet Européen ESPRIT II 2172 a pour but d'intégrer aux systèmes de contrôle-commande de nouvelles fonctionnalités concernant la maintenance et la gestion technique. La conception simultanée des trois pôles "contrôle, maintenance et gestion technique" a pour but d'améliorer la productivité et de réduire les coûts des matériels, de maintenance et de stocks [IUN 91].

Pour appliquer ce concept de CMMS, les travaux de la CIAME (Commission Industrie-Administration pour la Mesure) [CIA 88] ont abouti à l'élaboration de deux "livres blancs" regroupant les "réflexions et les besoins des utilisateurs de capteurs et d'actionneurs dans un environnement industriel".

Parallèlement, le projet DIAS (Distributed Intelligence Actuators and Sensors) [DIAS 89], a pour objectif un "actionnement et une mesure intelligente" pour améliorer la performance des instruments de terrain et assurer la cohérence des données d'une application.

Le projet DIAS définit trois classes de composants (cf. figure 1.4.):

- capteurs et actionneurs "intelligents",
- un nouveau système de maintenance informatisé réalisé grâce aux capteurs et actionneurs "intelligents",
- un système de contrôle et une nouvelle interface homme-machine pour les capteurs et actionneurs "intelligents".

L'ensemble des composants et des informations doivent être intégrés par un nouveau réseau de terrain : FIP, développé par une coopération franco/italienne. Le projet définit les "Companion Standard" : définition du format des données échangées sur le réseau de terrain pour la constitution des VMD (Virtual Manufacturing Device).

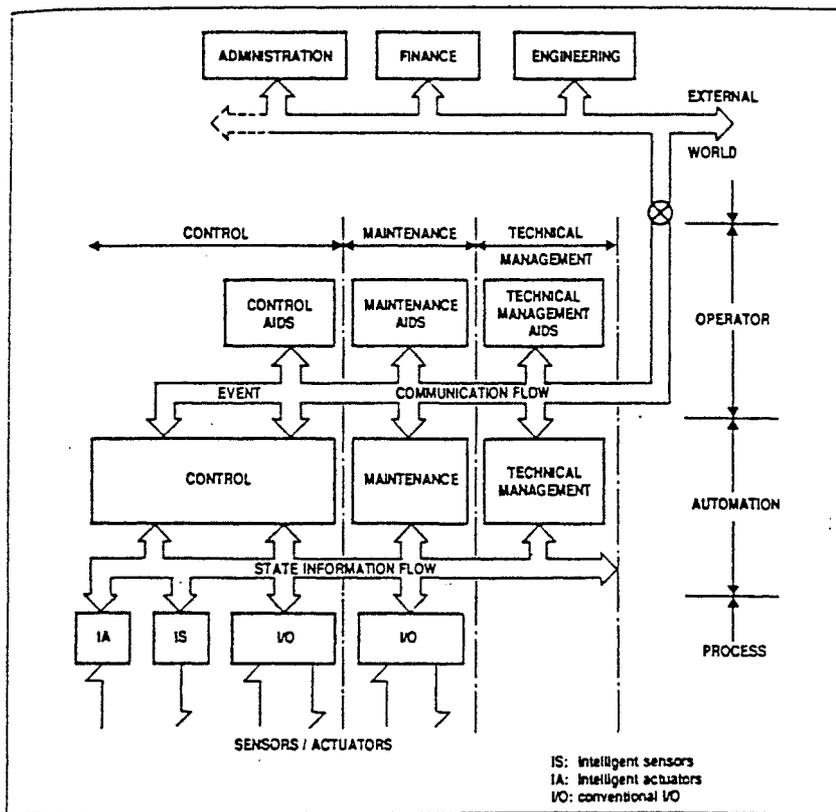
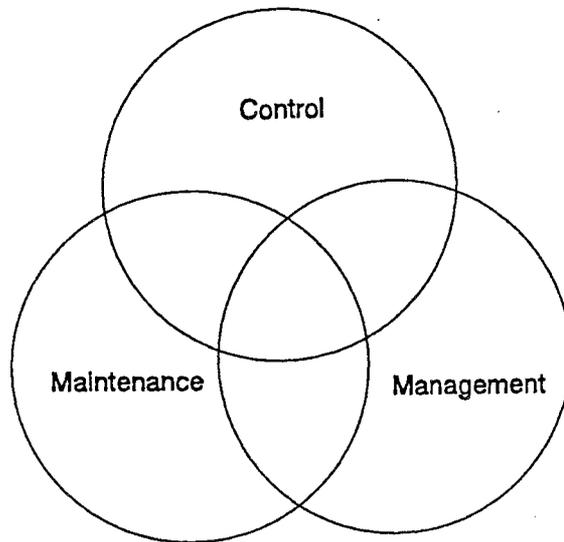


figure 1.4. Le concept de CMMS [DIA 89]

L'ensemble de ces concepts (CMMS, DIAS), notamment la distribution de l'intelligence au niveau de l'actionnement et de la mesure seront appliqués à la vision.

#### **2.7.4. Design Management and Manufacturing Système (D.M.M.S.)**

L'objectif du concept de DMMS est de faire cohabiter dans une même architecture les fonctions de conception de l'entreprise avec les fonctions de fabrication, autour du système d'information. L'objectif primordial de cette approche est de faire communiquer différents métiers (notamment les mécaniciens et les automaticiens) et d'établir un lien "sémantique" entre les intervenants du système de production. Ce projet est actuellement à l'étude dans notre laboratoire [LOM 92] et devrait être supporté par la norme P.C.T.E. (Portable Common Tool Environment) autour de la notion de "Poste de Travail". Ces travaux suggèrent une extension en vision par la définition d'un "Poste de Travail de Vision".

Le projet CIM-ONE [SHE 90] poursuit les mêmes objectifs avec une approche basée sur une modélisation-objet des données réunissant l'ensemble des fonctions de l'entreprise autour d'un "modeleur commun". La modélisation objet facilite la représentation de la sémantique (attribut, relation, agrégation, généralisation, héritage, instanciation, spécialisation, méthode, inférence, contrainte ,etc ...) d'une application et permet de gérer la dynamique des instances (incomplétudes et incohérence des objets). Cependant le problème de la dynamique des classes (ou schémas) n'est pas encore résolu, de même que le problème de la dynamique des modèles ne l'est en relationnel.

#### **2.8. Prototype de validation du projet MRT**

L'objectif de ce projet [MRT 91] est l'étude et l'auto-contrôle d'une cellule flexible d'usinage ainsi que l'étude de son intégration dans un contexte de CFAO. Les travaux ont été réalisés par trois laboratoires universitaires français : le centre de recherche en automatique de Nancy (CRAN), le laboratoire d'automatique de Besançon (LAB) et le laboratoire de robotique et d'informatique industrielle de Nantes (LRII).

Ces recherches comportent trois axes :

- le système d'information et la gestion de la qualité,
- le pilotage et l'auto-contrôle,
- la communication.

Ce projet a donné lieu dans notre laboratoire (CRAN/LACN) à la réalisation d'un prototype de cellule flexible comprenant :

- des matériels d'usinage : NUM760, CHARLY-ROBOT, montage d'usinage,
- des matériels de vision : Carte EDGE 90, CAIMAN
- des réseaux : ETHERNET, MAP, BITBUS,
- une base de données technique : SQL SERVER.

La plupart des travaux de ce mémoire ainsi que [DOI 92] et [COU 91] ont fait l'objet d'une validation technologique sur ce prototype.

### **3. Les concepts d'intégration méthodologique**

Les différents modèles que nous avons vus précédemment (NIST, CAMI, ICAM) ne prennent pas en compte l'aspect **conception des systèmes CIM**. C'est pourquoi des recherches ont été menées visant à définir une approche globale pour la conception et l'implantation des systèmes CIM. L'essentiel de ces travaux a été effectué dans le cadre du programme européen ESPRIT et a donné naissance à différents projets : CIM-OSA (Computer Integrated Manufacturing Open System Architecture, projet ESPRIT 688), MMCS (Manufacturing Management & Control System, projet ESPRIT 418), IMPACS (Integrated Manufacturing Planning And Control System, projet ESPRIT 2338) et FOF (Factory Of Future).

#### **3.1. Le projet IMPACS [CHE 90]**

Le projet a défini une architecture CIM constituée d'un ensemble de modèles de référence. Ces modèles servent de base à l'implantation d'une application particulière. Les modèles de référence décrivent différents points de vue : fonctionnel, décisionnel, informationnel, physique. Ces points de vue sont

décomposés en trois types d'architecture de référence : conceptuelle, structurelle et réalisationnelle.

Le modèle conceptuel de référence de la vue fonctionnelle décrit les fonctions de l'entreprise (en formalisme IDEF0), le modèle conceptuel de référence de la vue physique décrit les ressources (techniques de simulation et formalisme IDEF0), le modèle conceptuel de référence de la vue décisionnelle décrit la structure du système décisionnel (Réseaux GRAI) et le modèle conceptuel de référence de la vue informationnelle représente les informations du système de production (en formalisme Entité/Relation). L'ensemble de ces modèles conceptuels de référence décrivent le "QUOI", les modèles structurels de référence décrivent le "COMMENT" en utilisant d'autres formalismes (modèle logique de données pour la vue informationnelle), les modèles réalisationnels de référence décrivent le "AVEC QUOI", exprimé en terme de matériel informatique, matériel de fabrication ...

Outre ces deux axes : axe de point de vue et axe des niveaux d'abstraction (conceptuel, structurel, réalisationnel), le cadre de modélisation IMPACS fournit un troisième axe : l'axe méthode (ou cycle de vie). La méthode est une démarche structurée permettant de concevoir une application particulière à partir des modèles conceptuels de référence. La méthode IMPACS est basée sur une approche "cycle de vie" composée de trois phases : analyse, conception et implantation, pendant lesquelles les modèles de référence sont particularisés sous forme de modèles cible propre à l'entreprise particulière basés sur une étude de l'existant.

Le cadre de modélisation IMPACS est donc un modèle à trois axes (cf. figure 1.5.) : axe des vues, axe des niveaux d'abstraction, axe de la méthode.

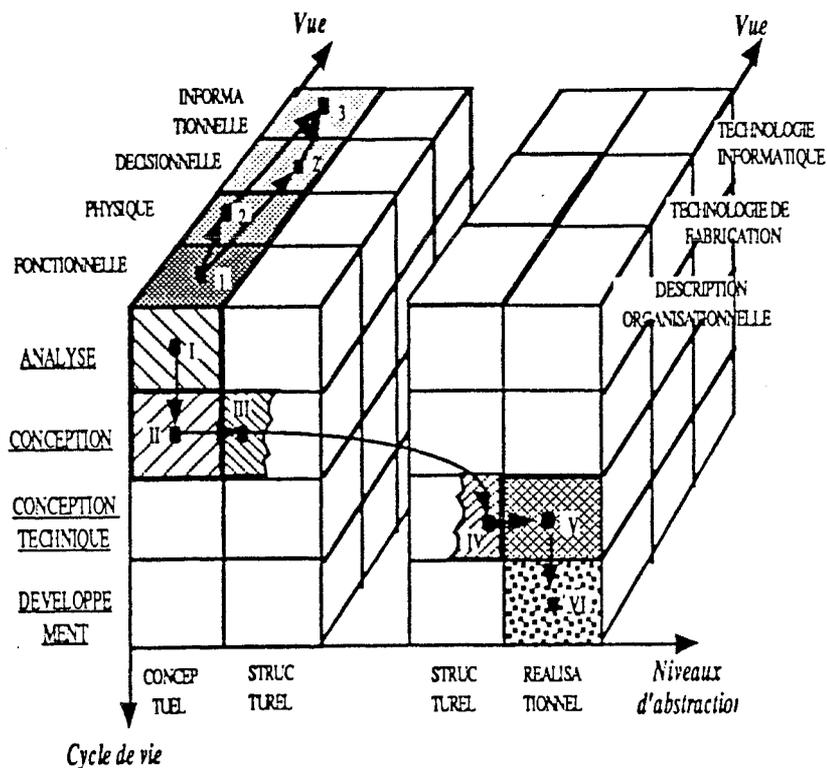


figure 1.5. Le cadre de modélisation IMPACS [CHE 90]

### 3.2. Le cadre CIM OSA [GAC 90]

Le consortium AMICE (European Computer Integrated Manufacturing Architecture) a défini l'architecture ouverte CIM-OSA et la méthodologie associée. Le modèle "cube" (cf. figure 1.6.) CIM-OSA est bâti sur trois principes :

- une approche de *réutilisation* ou "axe d'instanciation" qui suggère de construire le modèle particulier de l'entreprise à partir de modèles partiels, eux même exprimés en terme de construction générique de base,
- une approche *cycle de vie* ou "axe de dérivation" qui invite d'abord à modéliser les besoins de l'entreprise, puis les spécifications de conception et enfin la description de l'implantation,
- une approche *point de vue* ou "axe de génération" qui propose de modéliser suivant quatre vues : fonctionnelle, information, ressource et organisation.

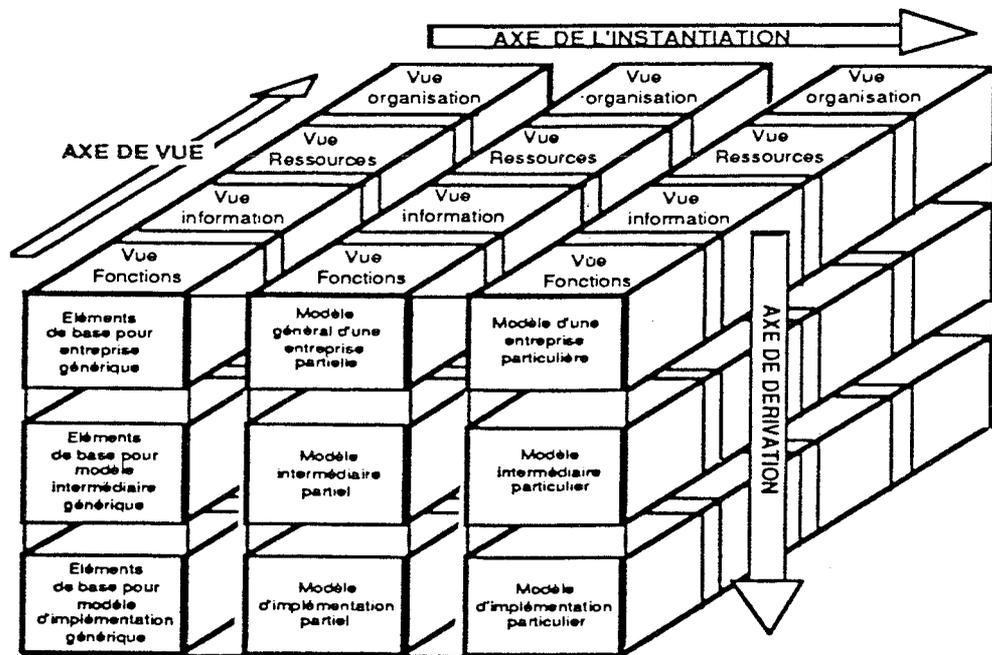


figure 1.6. Le cadre de modélisation CIM-OSA [GAC 90]

L'objectif de ces cadres de modélisation est de permettre à terme une approche semi-générative pour la conception d'une partie d'une application par encapsulation du savoir-faire tout en fournissant une démarche structurée de type cycle de vie.

### 3.3. Application au Génie Visionique

Nous cherchons à intégrer la fonction vision dans le Système Intégré de Production de l'entreprise. Notre Modèle Conceptuel de Référence Fonctionnel (cf. figure 1.3.) nous impose la décomposition de notre étude en trois domaines : le système physique, le système de gestion et le système d'informations industrielles. A l'image de la méthodologie CIM-OSA et IMPACS, nous chercherons à étudier l'intégration du "domaine vision" dans le S.I.P. selon les points de vue fonctionnel et informationnelle (axe génération).

#### 3.3.1. "Vue fonctionnelle" du "domaine vision"

Le but de la "vue fonctionnelle" du domaine vision est de fournir des outils et des méthodes permettant de décrire la fonctionnalité, la structure de contrôle

et le **comportement** du système de vision. Nous devons donc définir des modèles et architectures de référence (axe instanciation) permettant d'en déduire des modèles particuliers. D'autre part il nous faut définir une démarche de type "cycle de vie" (axe dérivation") permettant à partir de l'expression des besoins d'arriver à la description concrète d'application de vision. En somme nous devons trouver des outils et méthodes permettant d'**encapsuler le "savoir-faire fonctionnel" du Génie Visionique**. Cette "vue fonctionnelle" sera traitée dans l'étude de l'intégration du système de vision dans le système physique et dans le système de gestion du S.I.P.

### **3.3.2. "Vue informationnelle" du "domaine vision"**

Le but de la "vue informationnelle" est de permettre la **spécification du Système d'Informations Industrielles du S.I.P limité au domaine de la vision**. Nous appliquons le modèle "Entité/Association étendu" qui respecte le concept de "dérivation" à travers ses trois niveaux de décomposition d'une application : conceptuel, organisationnel et logique. Les modèles que nous spécifions sont les plus généraux possibles et peuvent être instanciés à des cas particuliers d'application de vision (axe instanciation). La "vue information" du "domaine vision" est appliquée à tout le S.I.P. Notre objectif est là aussi d'**encapsuler notre savoir-faire informationnel du "domaine vision"** en vue d'une instanciation ultérieure. Les modèles conceptuels qui en résultent sont donc très généraux (modèles de référence) et peuvent être réutilisés pour définir par instanciation des "*modèles conceptuels cible*" (voir IMPACS). **Notre objectif dans ce mémoire est de définir l'ensemble des Modèles Conceptuels de Données de Référence du "domaine vision" dans l'entreprise.**

### **3.3.3. Situation de nos travaux dans le modèle CIM-OSA**

L'ensemble des modèles (fonctionnels et informationnels) que nous établissons dans ce mémoire sont conçus de façon à être applicables à la majorité des systèmes de contrôle sans contact. Nos modèles sont donc des modèles de référence; ils servent de base pour la réalisation d'un cas particulier d'application. Nous nous situons donc en haut de l'axe de dérivation et à gauche

de l'axe d'instanciation (cf. figure 1.7.). Nos modèles sont instanciables et sont les points d'entrée du cycle de vie de conception.

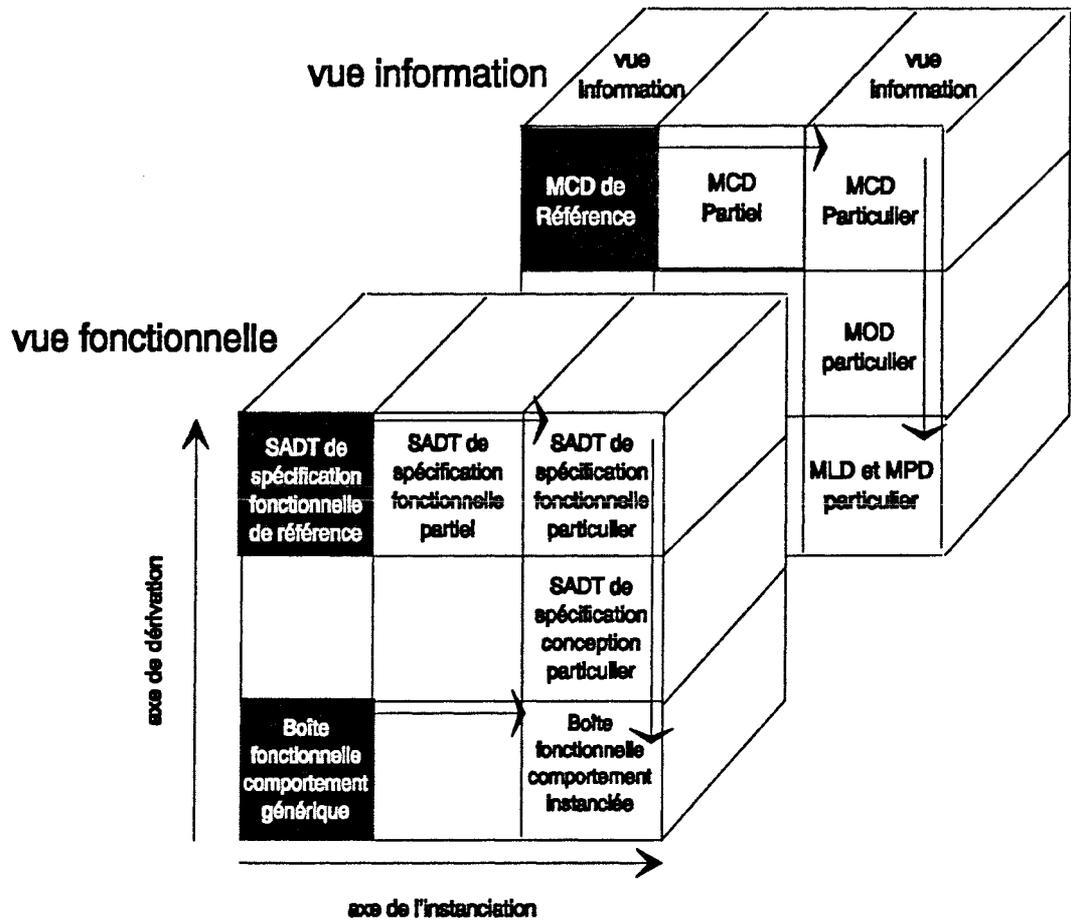


figure 1.7. La situation de nos travaux dans le modèle CIM-OSA

#### 4. Les outils et méthodes utilisés

Tout au long du mémoire nous utilisons des formalismes et leurs outils associés. Nous désirons les présenter maintenant dans un souci de clarté.

##### 4.1. Le formalisme Entité/Relation Etendu, l'outil MEGA

MEGA [MEG 89] est un Atelier de Génie Logiciel d'assistance à la conception et au prototypage des Systèmes d'Information (SI). Il est construit

autour du modèle "Entité/Association étendue" et permet la modélisation d'un Système d'Information suivant trois niveaux (conceptuel, organisationnel et logique). A chaque niveau correspondent des modèles décrivant des points de vue différents (activités, communications, données, traitements). Nous ne nous intéresserons qu'aux données :

- le niveau *conceptuel* est le niveau de description du fonctionnement cible de l'entreprise (le quoi). Le Modèle Conceptuel de Donnée (MCD) définit les données requises par l'activité des intervenants, en particulier pour les échanges d'informations, le MCD est représenté en formalisme Entité/Relation,

- le niveau *organisationnel* représente le fonctionnement de l'entreprise. Le Modèle Organisationnel de Données (MOD) décrit les données en terme d'autorisation d'accès, de répartition géographique, de fraîcheur d'information et de système de mémorisation, le MOD est représenté en formalisme Entité/Relation,

- le niveau *logique* permet de spécifier les outils informatiques mis à la disposition de l'utilisateur. Le Modèle Logique de Données (MLD) est la transformation en formalisme relationnel d'un modèle représenté en formalisme Entité/Relation.

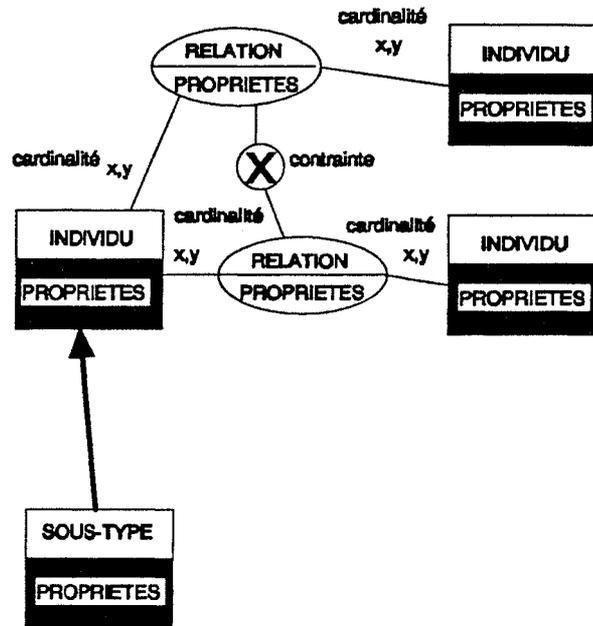
Le formalisme Entité/Relation étendu [TAB 86] permet la représentation des données. Il est composé de plusieurs éléments types (cf. figure 1.8.) :

- l'*individu* : un *individu* est une chose, un concept; il représente une donnée pure et est dessiné sous forme d'un rectangle; un *individu* est caractérisé par ses *propriétés*,

- la *relation* : une *relation* relie des *individus*; en effet, certaines données établissent des liens particuliers avec d'autres données; une *relation* est dessinée sous forme d'un ovale; une *relation* peut avoir des *propriétés*,

- le *sous-type* : un *sous-type* est un *individu* qui hérite des *propriétés* de l'*individu* dont il est le *sous-type* et qui peut avoir ses propres *propriétés*; le *sous-type* se représente par une flèche allant de l'*individu sous-type* à l'*individu* supérieur hiérarchiquement; il s'agit d'un héritage statique,

- les *contraintes* : une *contrainte* est une "obligation" que doivent respecter les données auxquelles elle s'applique; une *contrainte* se traduit par un contrôle à effectuer lors des traitements de mise à jour des données,



*figure 1.8. Le formalisme Entité/Relation étendu*

- les *règles* : une *règle* permet de contraindre la valeur d'une information par application d'une équation, d'une d'inéquation ou d'un algorithme,

- les *cardinalités* : une *cardinalité* est une *contrainte* appliquée à une *patte* reliant un *individu* à un autre *individu* via une *relation* (il y a deux *cardinalités* par *relation*). Les *cardinalités* caractérisent le type de relation entre deux ensembles d'*occurrences* (bijection, surjection, injection, etc ...) :

- 0,1 : peut être,
- 0,N : peut-avoir,
- 1,1 : doit être,
- 1,N : doit avoir.

L'ensemble de ces éléments types permet de modéliser une partie de la sémantique (savoir-faire) de l'application qui joue un rôle très important en productique. Les *règles, contraintes et sous-types* sont finalement implantés dans une base de données relationnelle sous forme de traitements.

L'Atelier Logiciel MEGA permet la création de MCD et MOD étendus, il fournit en outre la possibilité de traduire les MOD en MLD. Les concepts "d'individu" et de "relation" utilisés dans le formalisme "Entité/Relation" correspondent aux segments et attributs du **méta-modèle** de MEGA (cf. figure 1.9.). Ce méta-modèle représenté sous la forme d'un MCD est implémenté physiquement dans le SGBD de MEGA pour constituer la base de données de mémorisation des applications menées avec l'Atelier Logiciel. La description de ce méta-modèle est elle aussi représentée sous forme d'un MCD appelé **méta-méta-modèle** composé de trois individus : le segment, le lien, l'attribut.

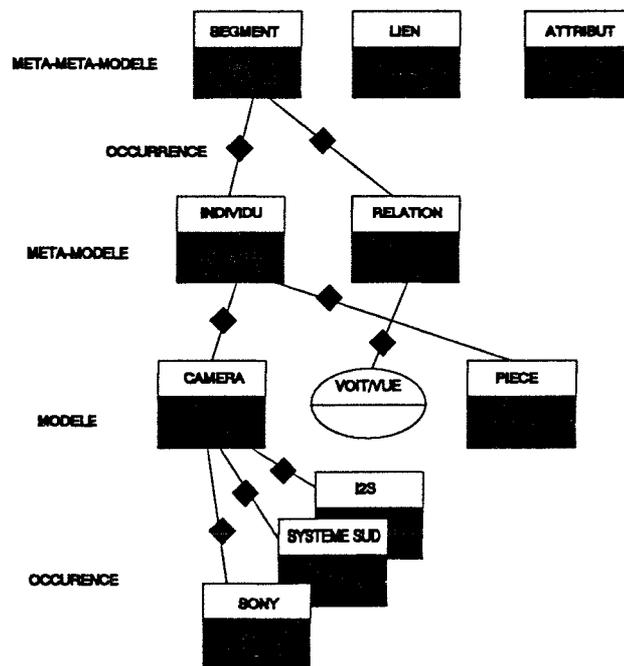


figure 1.9. Concepts de MEGA

#### 4.2. Le formalisme SADT, l'outil ORCHIS

SADT [IGL 89] est une méthode de spécifications fonctionnelles hiérarchique descendante, elle conduit à la réalisation de un ou plusieurs modèles dans lesquels l'accent est porté sur le QUOI, c'est à dire sur les fonctions que le système doit remplir. Ces fonctions sont décrites de manière conceptuelles, indépendamment de la manière de les réaliser.

Nous utilisons l'outil ORCHIS qui fournit un éditeur de SADT dans le cadre d'une démarche de conception de la commande des systèmes automatisés.

#### 4.3. Le formalisme GRILLE, l'outil ORCHIS-GRILLE

Le formalisme grille [LHO 85] intervient dans une démarche descendante de conception de Machine et Systèmes Automatisés de Production basée sur la méthode SADT. La méthode se compose de trois étapes :

- étape 1 : décomposition fonctionnelle (cf. figure 1.10.) en formalisme SADT (avec ORCHIS) du processus en une série de sous-processus auxquels on associe systématiquement une fonction de "contrôle-commande". On doit retrouver cette fonction à chaque niveau de décomposition,

- étape 2 : détermination des fonctions de "contrôle-commande" par utilisation d'un modèle "standard" de référence [LHO 85], indépendant de toute application. Ce modèle "standard" de référence pour la conception de la partie commande est (cf. figure 1.11.) défini comme étant la réunion de trois fonctions standards chacune redécomposée :

- piloter : suivre, optimiser, régler,
- conduire : percevoir, réfléchir, agir,
- coordonner : surveiller, réagir, commander,

l'outil associé ORCHIS-GRILLE [PAN 91] informatise les grilles (cf. figure 1.12.) de description des fonctions de contrôle-commande préconisées par [LHO 85] en liaison directe avec les résultats fournis par ORCHIS (SADT).

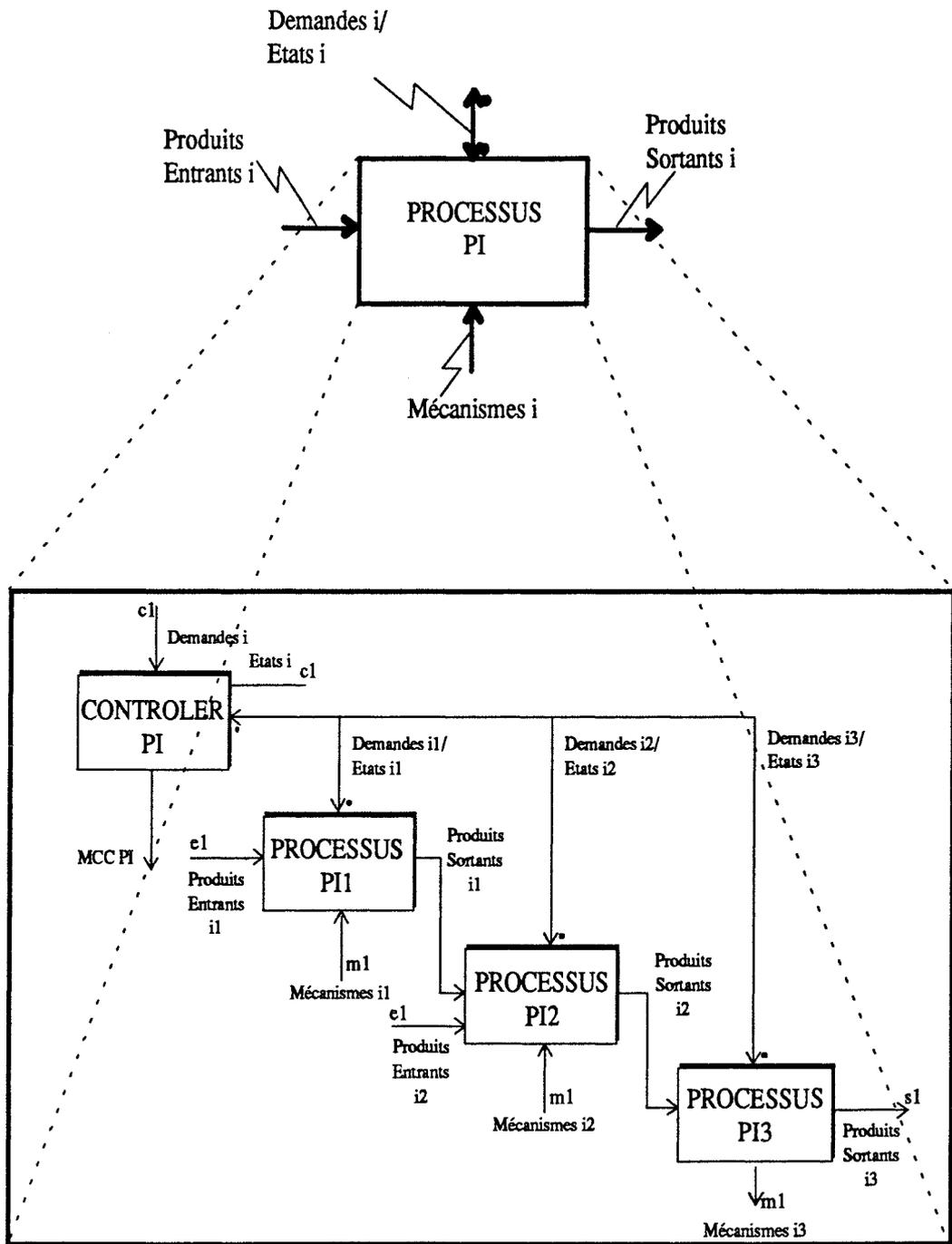
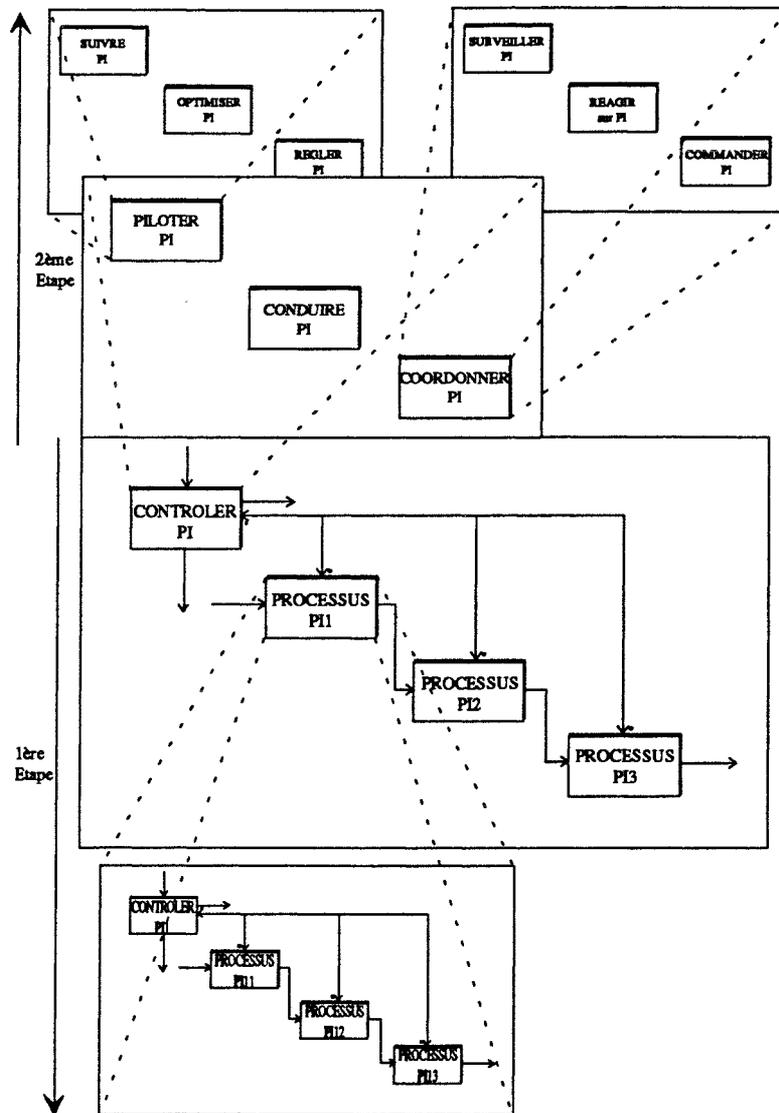


figure 1.10. Décomposition d'un processus



*figure 1.11. Processus de décomposition d'une fonction de contrôle*

En outre ORCHIS-GRILLE permet de créer ses propres modèles de référence particuliers; le formalisme "grille" permet donc de décrire les interfaces d'entrée/sortie des différentes fonctions composant la commande que l'on conçoit, ces interfaces sont réutilisées par l'outil SPEX [TIX 89] pour le prototypage de leur comportement,

- étape 3 : description complète de la partie commande par prototypage sur SPEX en langage métier (grafcet, logigramme, langage C) du comportement de chaque fonction conçue avec l'outil ORCHIS-GRILLE.

DIAGRAMME : CONTROLER P1				GRILLE : MCC P11					A.1
Informations	CONTROLER			SOUS-PROCESSUS	SOUS-PROCESSUS	SOUS-PROCESSUS	SOUS-PROCESSUS	SOUS-PROCESSUS	
	CONTROLER SUPERIEUR	PLOTER	CONDUIRE	COORDONNER	No 1	No 2	No 3	No 4	
CONTROLER SUPERIEUR	CONTROLER P1	1	3	5					
C O N T R O L E R	PLOTER	2	7	9	13	15	19	25	31
	CONDUIRE	4	CONDUIRE	11	17	21	27	33	37
	COORDONNER	6	10	12	23	29	35	39	41
SOUS-PROCESSUS		14	18	24	P111				
SOUS-PROCESSUS		16	22	30		P112			
SOUS-PROCESSUS		20	28	36			P113		
SOUS-PROCESSUS		26	34	40				P114	
SOUS-PROCESSUS		32	38	42					P115

figure 1.12. Grille associée à une fonction de contrôle-commande

#### **4.4. Les formalismes et l'outil SPEX**

L'objectif de SPEX est de permettre la réutilisabilité d'objets d'automatisation en préservant le savoir-faire et les formalismes courants des automaticiens (grafcet, logigrammes, relais, langage C). Il permet :

- La description de comportements génériques (Boîtes Fonctionnelles) (cf. figure 1.13.),
- la composition d'applications à partir d'exemplaires de Boîtes Fonctionnelles (Diagrammes Fonctionnels),
- la mémorisation et la duplication de ces composants,
- leur simulation avant codage.

SPEX est un environnement pour la spécification et la conception de Systèmes Automatisés de Production qui met en oeuvre quelques principes importants du Génie Logiciel pour assister l'Automatiseur et le Génie Automatiseur. Il fournit un ensemble de moyens et démarches permettant une amélioration de la qualité de l'automatisation par une meilleure réutilisation du savoir-faire de l'entreprise où d'un domaine particulier. Il permet de modéliser la partie commande d'une application, mais aussi d'émuler la partie opérative.

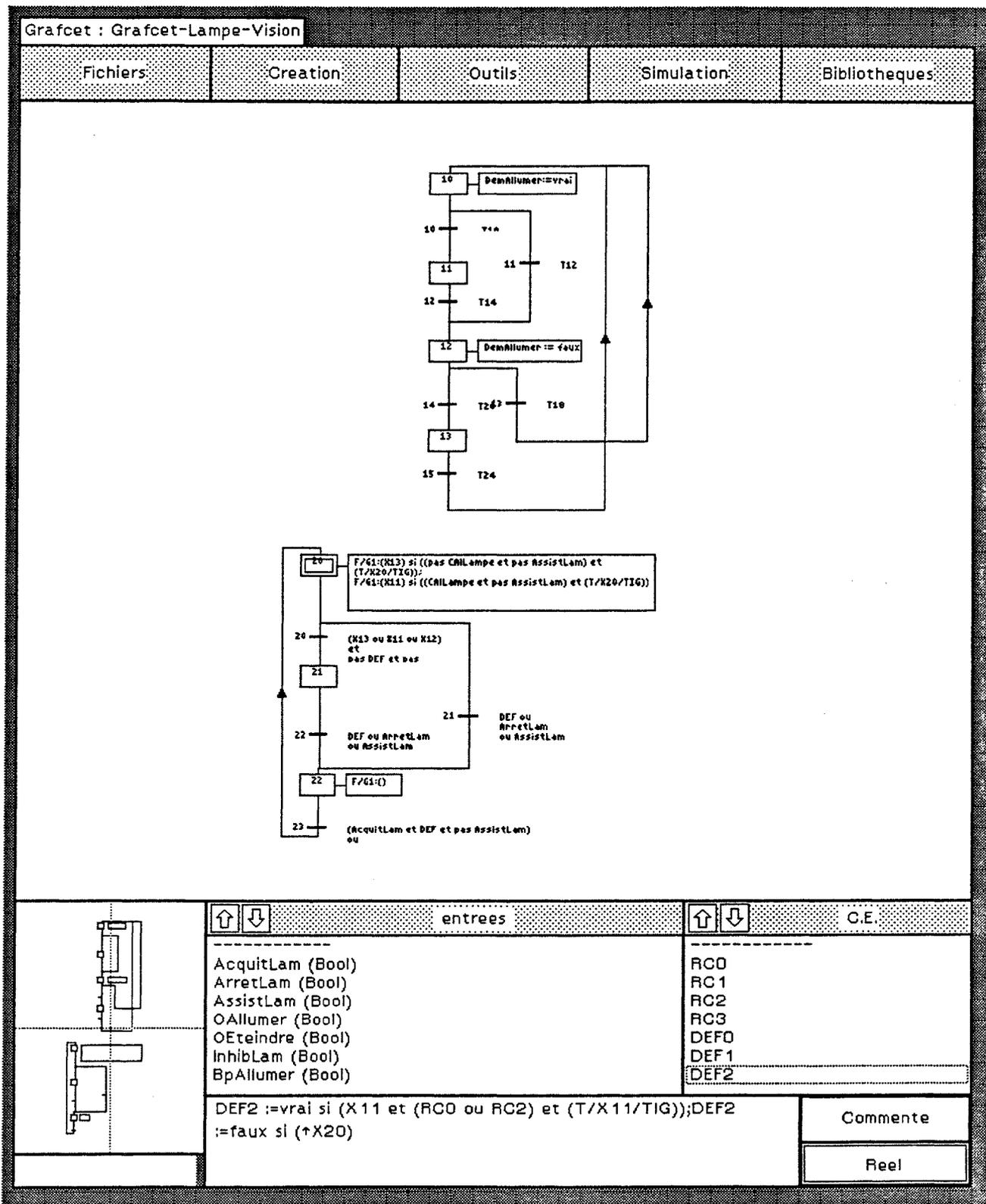


figure 1.13. Vue interne de la boite fonctionnelle : commander une lampe

## Chapitre 2

# Vision et informations techniques

1. Problématique .....	1
2. Les éléments de l'acquisition d'image.....	2
2.1. La source .....	4
2.2. L'objet .....	5
2.3. Le récepteur.....	7
3. Les éléments de l'interprétation d'image.....	10
3.1. Analyse de différents systèmes de vision.....	11
3.1.1. Intermediate Symbolic Representation ISR [BRO 89].....	12
3.1.2. Knowledge-Based Computer Vision [GOO 89].....	13
3.1.3. Thin Line Code (TLC) [JAG 89] .....	13
3.1.4. Powervision [CON 89].....	14
3.1.5. Base de données pour la représentation d'objets plats [QIN 88].....	15
3.1.6. Base de données image pour une interprétation sémantique [PIZ 89].....	16
3.1.7. Conclusion sur l'analyse de différents systèmes de vision .....	16
3.2. Traitements et données .....	17
3.2.1. La numérisation et les prétraitements.....	20
3.2.2. La segmentation et l'interprétation.....	20
3.2.3. La représentation des traitements.....	22
3.2.4. La représentation du monde visuel.....	23
3.2.5. Les programmes et les données utilisés par les traitements .....	25
3.2.6. Les données issues du traitement .....	26
4. L'individu SYSTEME DE VISION .....	28
5. Application : système de vision matricielle .....	28
5.1. Les optiques.....	29
5.2. Les capteurs .....	29
5.3. Les éclairages .....	29
5.4. Les scènes.....	29
5.5. Les traitements.....	31
5.5.1. Les prétraitements .....	31
5.5.2. La segmentation .....	32
5.5.3. L'interprétation .....	32
5.6. Conception du Système d'Information intégrant le contrôle qualité par vision.....	34
5.6.1. MCD de Référence .....	34
5.6.2. MCD partiel .....	34
5.6.3. MOD particulier .....	36
5.6.4. MLD particulier.....	37
5.6.5. Génération du code SQL.....	38

## Chapitre 2

# Vision et informations techniques

### 1. Problématique

Ce chapitre a pour but l'étude des informations nécessaires au fonctionnement d'un système de vision. Cette étude a trois objectifs :

- une meilleure connaissance du procédé de vision,
- générer une partie du MCD de référence du "domaine vision",
- permettre la génération d'un Système d'Informations Industrielles particulier intégrant la visionique

La démarche systémique nous impose de partir de "l'existant" afin d'en extraire un "invariant conceptuel", c'est pourquoi nous avons réalisé une bibliographie dont le thème est : "vision et informations techniques". Nous utilisons cette étude comme base de l'état de l'art en vision par ordinateur. L'étude de chaque élément d'un système de vision nous amène à extraire les données correspondantes, utiles à la fois à la fonction de "gestion technique" et de "fabrication".

Ce chapitre est divisé en deux parties : "les éléments de l'acquisition d'image" et "les éléments du traitement d'image"; en effet comme le fait remarquer [MEZ 88] :

*"Un système de Vision par ordinateur doit posséder deux parties, une correspondant à la phase de perception, il s'agit de remplacer l'oeil par un système similaire qui permette de <<photographier>> des images avec le maximum de renseignements subtiles, l'autre, l'interprétation et la décision, où tout se complique, car des multitudes de procédés existent".*

Par essence même le système de vision permet la séparation des données et des traitements, l'acquisition des données faisant intervenir l'essentiel des parties opératives du système de vision. L'automatisation de celui ci consiste donc surtout à maîtriser la prise d'image et le stockage de celle ci. Son interprétation ne faisant intervenir que du traitement d'informations.

En conclusion de ce chapitre nous présenterons de façon descriptive notre système de vision et nous montrerons comment instancier une partie du MCD de Référence Vision dans une application particulière.

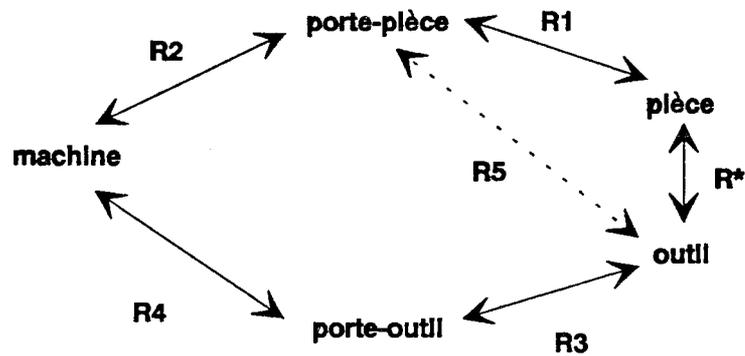
## 2. Les éléments de l'acquisition d'image

La majorité des capteurs sans contact fonctionnent de la manière suivante :

Une source émet un signal (lumière naturelle, lumière structurée, ultra-son, champ magnétique...), ce signal entre en contact avec l'objet à "observer" et est renvoyé vers un récepteur (capteur). Ce schéma de fonctionnement est valable aussi bien pour un système de vision matricielle que pour un système plus complexe de télémétrie laser ou ultra-son.

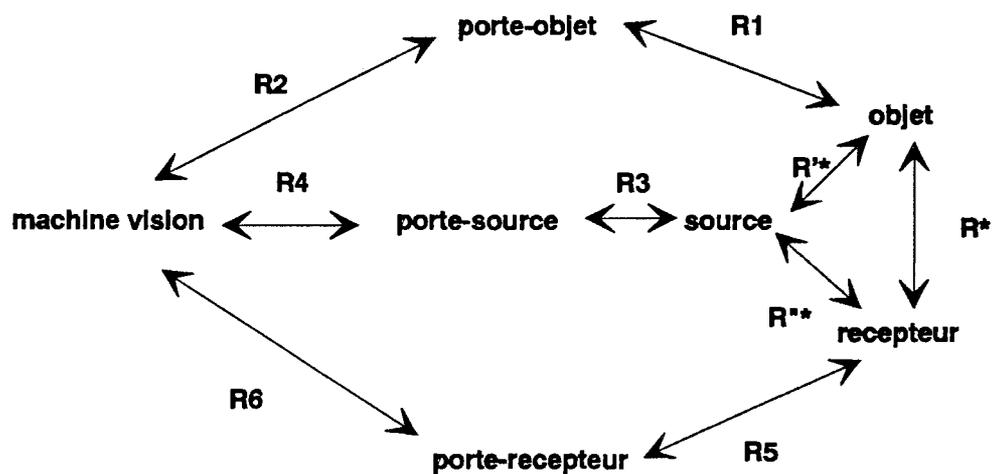
On distingue donc toujours trois intervenants : la source, le récepteur et l'objet (qui ne fait pas partie du système de vision, mais qui participe forcément à son fonctionnement).

Nous pouvons comparer le processus de contrôle au processus d'usinage. La figure 2.1. représente le P.E.U. (Processus Élémentaire d'Usinage). *"Ce schéma relationnel exprime les relations entre les différents intervenants du P.E.U. La relation  $R^*$  caractérise la liaison entre l'outil et la pièce de laquelle dépend directement la qualité du travail réalisé sur le P.E.U. Dans cette liaison, l'outil contribue à la formation du copeau et à la géométrie des surfaces"* [MAT 86]. La description de la relation  $R^*$  permet de choisir l'outil.



*figure 2.1. Processus élémentaire d'usinage*

A l'image du P.E.U, le processus de contrôle sans contact peut se schématiser de manière relationnelle (cf. figure 2.2.), il est composé de 7 éléments de base : la machine vision, le porte-récepteur, le récepteur, le porte-source, la source, le porte-objet, l'objet.



*figure 2.2. Processus élémentaire de contrôle sans contact*

La relation  $R^*$  se décompose en deux relations,  $R'^*$  et  $R''^*$ ; ces deux relations contribuent à la formation du signal capté (signal tout ou rien, image, mesure de distance, etc... ). Ce schéma constitue notre point de départ pour l'étude des informations du système de vision. Notre objectif est de spécifier l'ensemble de ces relations afin de parvenir au choix des instruments de vision à partir de la définition de "l'objet".

## 2.1. La source

Les différentes sources de signaux utilisés par les capteurs sans contact sont :

- champ magnétique (capteur par induction),
- champ électrique (capteur capacitif),
- onde électromagnétique (caméra classique et infrarouge, rayons x),
- onde sonore (capteur ultra-son).

Dans une application de vision, l'éclairage joue un rôle primordial. Tous les spécialistes s'accordent pour reconnaître son rôle capital. "*Le coût d'un bon éclairage est loin d'être négligeable et peut dans certain cas correspondre à plus de 20% de l'investissement global*" [MEZ 88].

D'une manière générale, la source la plus utilisée est la lumière visible. On distingue trois types d'éclairages :

- éclairage *stroboscopique* : lorsque l'objet est en mouvement, l'utilisation d'un stroboscope synchronisé permet d'éviter le phénomène de flou photographique (l'image de l'objet "avant" la prise d'image n'existant pas)
- lumière *continue structurée* : ce sont les lasers; ce type de système est utilisé pour obtenir une information 3D de l'objet observé
- lumière *continue non structurée* : l'éclairage est assuré par les lampes du commerce ou par des produits spécialement conçus pour ces applications; il est rare de travailler en lumière naturelle, les spécialistes recréent pour chaque application un environnement entièrement dédié à celle ci pour s'affranchir au maximum des variations externes.

L'*individu* SOURCE ELECTROMAGNET (cf. figure 2.3.) est composé d'un ensemble de composants spécifiques. Par exemple une simple lampe est constituée par :

- un pied de lampe,
- un raccord articulé avec le projecteur,
- un projecteur,
- un parapluie diffuseur,

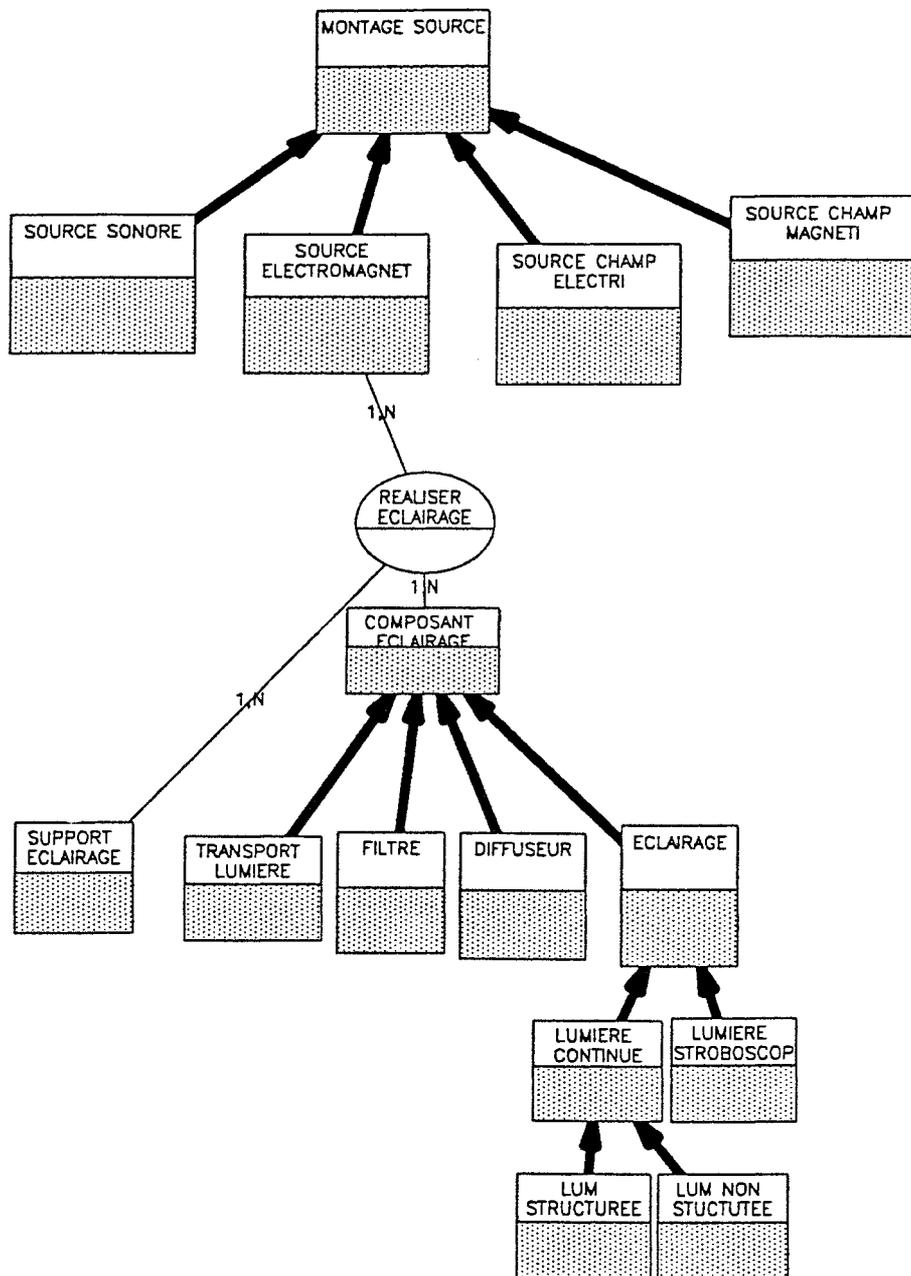


figure 2.3. Extrait du MCD de référence de vision : MONTAGE SOURCE

## 2.2. L'objet

Le rôle de l'objet est primordial : renvoyer le signal modifié vers le récepteur;

Nous verrons au chapitre IV comment il est possible que la conception d'un système de vision entraîne une rétroaction sur la conception du produit, de façon à ce que celui ci soit en mesure de renvoyer un signal "recevable". Vu sous cette

optique, l'objet fait bien partie intégrante du système de vision. Et du même coup, le système de vision devient un élément intégré du système de production.

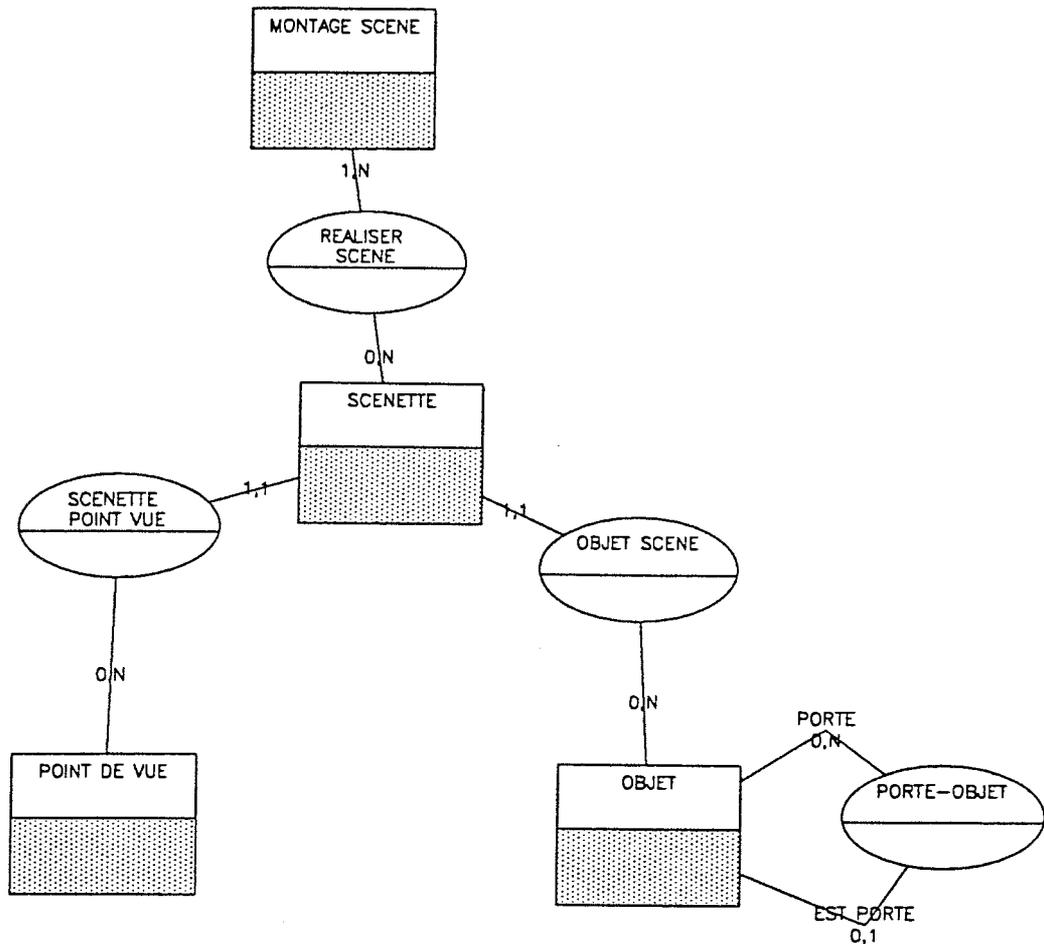


figure 2.4. Extrait du MCD de référence de vision : **MONTAGE SCENE**

L'objet influence donc le fonctionnement du système par sa forme, par sa couleur, par son contraste avec le fond, par sa matière. L'expert qui met au point un système de vision peut remettre en cause l'objet ainsi que tout l'environnement de l'application (pièce, porte-pièce, etc ...).

La scène est composée de tout ce qui est susceptible de renvoyer le signal vers le récepteur. Nous décomposons la scène en un *individu* OBJET qui peut être porté par un autre "objet" (*relation* PORTE-OBJET).

L'*individu* POINT DE VUE définit la position relative entre le capteur et l'objet observé (l'attitude). Dans de nombreux cas, les caractéristiques mêmes de la forme à observer impose un certain point de vue incontournable. Si l'encombrement de la scène (dans une machine outil) ne permet pas le positionnement d'une optique selon ce point de vue, il est inutile de poursuivre les recherches.

### 2.3. Le récepteur

La technologie du capteur dépend du signal porteur de l'information à capter. Nous effectuons une classification sur la dimension de l'information captée :

- *dimension nulle* : ce sont les capteurs de type "tout ou rien" capables de détecter la présence d'un objet dans leur proximité (capteur inductif, capteur capacitif),
- *dimension 1* : ce sont les capteurs linéaires d'une résolution supérieur à 4000 pixels (le mot "pixel" vient de l'anglais *picture element*). Ils sont utilisés dans les problèmes d'inspection où la précision de mesure est un facteur important et dans les cas où les objets sont en mouvement (tapis roulant),
- *dimension 2* : ce sont des systèmes qui exploitent des caméras matricielles (vidicon ou CCD); ils offrent une moins grande résolution (512\*512) que les systèmes linéaires. Quelques constructeurs proposent des caméras CCD couleur,
- *dimension 3* : ce sont des capteurs exploitant soit des principes télémétriques utilisant :
  - la mesure du temps de vol (d'une onde ultrason, d'un faisceau laser),
  - la triangulation par faisceau laser,soit des méthodes stéréoscopiques telles que la stéréovision (corrélation d'images fournies par des sources différentes) active ou passive (adjonction d'un faisceau de lumière structurée pour obtenir un plan laser). Ces systèmes fournissent une cartographie 3D, c'est à dire les coordonnées cartésiennes d'un ensemble d'éléments (points, segments de droite).

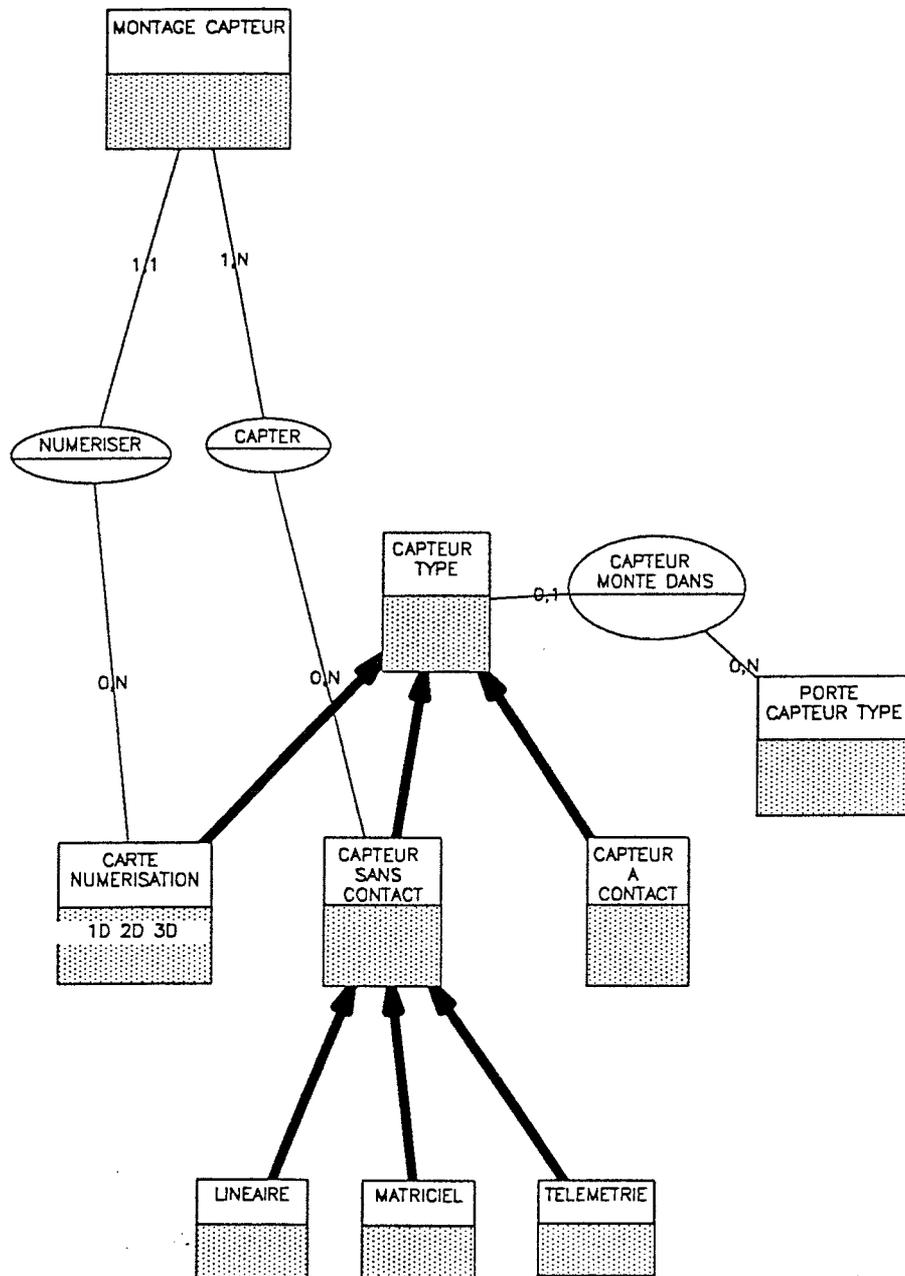


figure 2.5. Extrait du MCD de référence de vision : MONTAGE CAPTEUR

Voici une liste non exhaustive [BES 88] :

- Capteur de proximité,
- capteur par induction,
- capteur par effet-hall,
- capteur capacitif,
- capteur à ultrasons,

- capteur de distance par télémétrie par triangulation,
- capteur de distance par télémétrie par temps de vol,
- capteur de distance par télémétrie par laser,
- capteur de distance par télémétrie par ultrasons,
- caméra à tube Vidicon
- caméra à matrice CCD
- caméra linéaire CCD
- caméra couleur CCD
- capteur d'image tridimensionnel par télémétrie-laser stéréoscopique,
- etc...

L'*individu* MONTAGE CAPTEUR regroupe l'ensemble physique constitué du capteur (caméra + porte-caméra) et du système auquel il est relié (carte de numérisation) (cf. figure 2.5.).

Une *occurrence* de l'*individu* PORTE-CAPTEUR peut être un axe d'une machine outil qui permettrait de déplacer la caméra à volonté dans l'espace de travail de la machine. Le déplacement physique de la caméra est dans ce cas réalisé par l'exécution d'un programme de commande numérique.

L'*individu* CARTE DE NUMERISATION regroupe les *occurrences* des différents systèmes de numérisation utilisés par l'industriel. Par exemple :

- carte EDGE 90,
- carte MATROX,
- carte ELECTRONIQUE LYONNAISE,
- carte spécifique développée en laboratoire,
- carte pour caméra linéaire I2S,
- système cible AI VISION (reconnaissance de caractères),
- système cible caméra EIA,
- armoire de commande robot vision AVICOR d'ITMI
- etc...

L'*individu* MONTAGE OPTIQUE regroupe les éléments physique du système optique. Un système optique est constitué d'un ensemble d'éléments optiques (composants optiques) assemblés (cf. figure 2.6.).

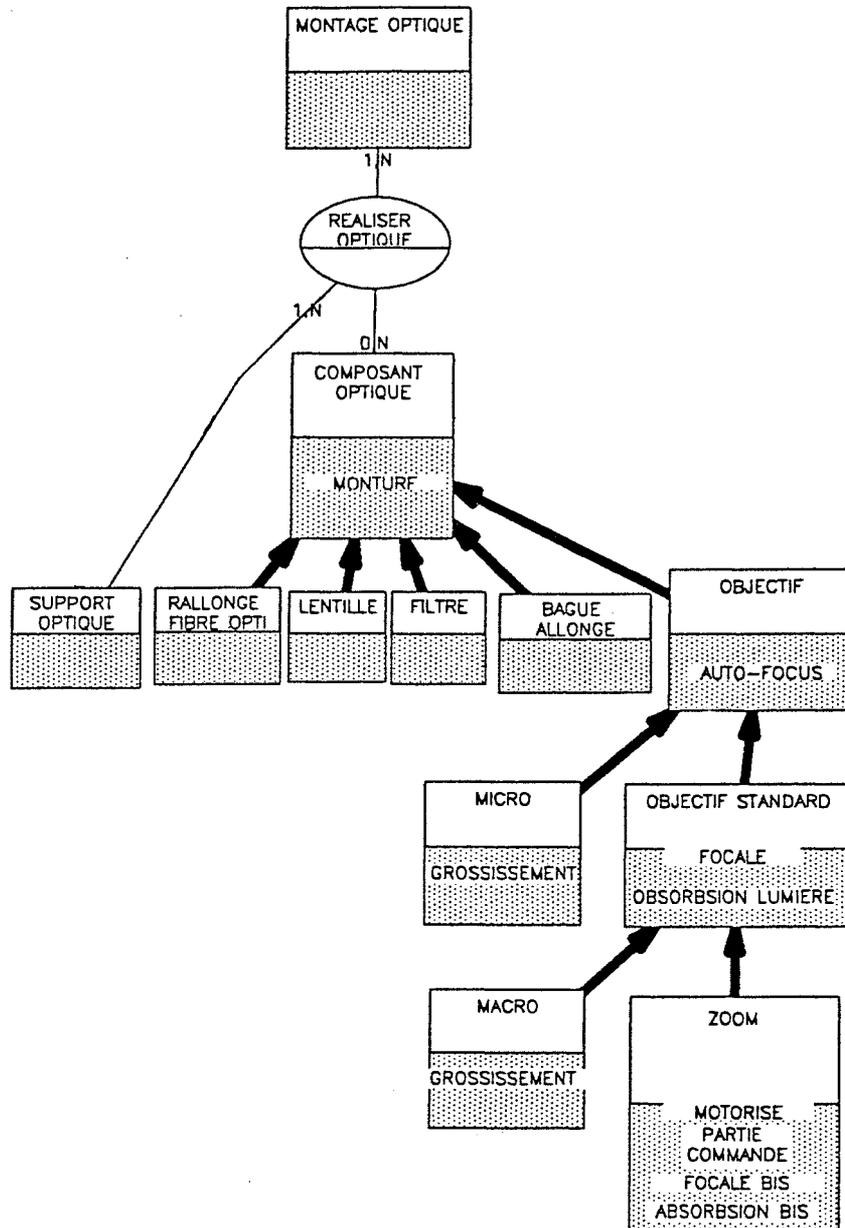


figure 2.6. Extrait du MCD de référence de vision : **MONTAGE OPTIQUE**

### 3. Les éléments de l'interprétation d'image

"Tout système de perception visuelle pour machine doit pouvoir former, analyser et interpréter des images. La formation de l'image est directe : l'ordinateur reçoit un ensemble de mesures de la quantité de lumière réfléchie

*dans les yeux-caméra du système à partir de la surface des objets. L'ordinateur pourrait se contenter de traduire ces mesures en image, mais la machine n'identifierait pas l'objet. Le système examine donc les nombres fournis, grâce à des algorithmes lui permettant de repérer les zones où l'intensité change brusquement . Le but de cette analyse : extraire des descriptions susceptibles d'utilisation dans le processus d'identification" [TIM 88].*

Nous avons réalisé une étude bibliographique volontairement orientée "données".

### **3.1. Analyse de différents systèmes de vision**

Un certain nombre de recherches ont été menées sur le thème "vision et données". La plupart de ces travaux ont été effectués dans une optique purement vision et débouchent finalement sur la spécification de systèmes simplement plus ergonomiques, plus faciles à mettre en œuvre, plus accessibles à l'utilisateur [GOO 89][GRO89][BRO89][CON89][JAG89].

Aucun type de base de données (relationnelle ou orientée objets) ne donne entièrement satisfaction aux concepteurs de base de données vision; la nature hiérarchique des images en vision artificielle (pixels, régions...) ne favorise pas l'emploi de BD relationnelles [JEA 87], pourtant, *"la modélisation de relations sémantiques (relation à plat, non hiérarchique) entre objets est indispensable pour l'interprétation des images"* [GOO 89]. Dans une base de données image, les modèles d'objet à manipuler (reconnaître, localiser) sont parfois inconnus, ce qui pose un problème pour définir un type d'objet qui correspond exactement à la tâche d'interprétation de l'image, qui n'est pas connue à priori [GRO 89]. Malgré tout, la quasi totalité des systèmes de vision ainsi conçus sont bâtis sur des BD orientées objet.

Les machines de vision industrielles doivent répondre à des impératifs de temps de traitement qui font que l'utilisation d'une base de données lors du fonctionnement du système est pénalisante, pourtant on constate l'importance de la séparation des données et des traitements pour améliorer *"l'expandability of machine vision"* [MAS 89]. L'intégrabilité d'une machine de vision passe par une meilleure compréhension de ses données. Dans cette optique, les travaux cités

plus haut sont une source précieuse d'information concernant les données utilisées en vision. La pratique de langages orientés objet ainsi que l'utilisation unique de bases de données orientées objet n'amenant pas les chercheurs à modéliser de manière formelle leurs travaux (comme les méthodes entité-association le permettent), l'extraction sémantique de leur savoir-faire est délicate.

### 3.1.1. Intermediate Symbolic Representation ISR [BRO 89]

De nombreux niveaux de représentation et d'abstraction (Intermediate-Symbolic-Representation) sont nécessaires en vision pour représenter le monde (d'une image). ISR est organisé en trois niveaux conceptuels :

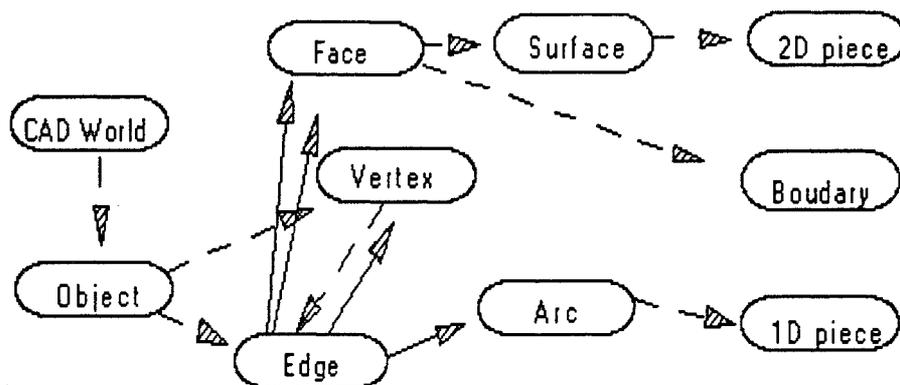
- **Bas** (image level) : régions, contours, ensembles de points, lignes, surfaces 3D. Ce niveau contient toutes les informations que l'on peut extraire sans connaissance particulière sur le contenu de l'image (ou du monde).
- **Intermédiaire** : les éléments de bas niveaux sont rassemblés pour former des entités symboliques appelées "tokens". Une hiérarchie peut exister à partir des différents "tokens".
- **High** (knowledge level) : la connaissance de "haut niveau" décrivant le monde est organisée en descriptions appelées "schemas". Ces "schemas" sont décrits par des réseaux relationnels (relational network) pour faciliter la reconnaissance. Chaque "schema" possède sa propre procédure qui recherche l'objet dans les niveaux bas et intermédiaire. Les "schemas" sont des entités actives (objets) qui communiquent de manière asynchrone entre elles.

Le modèle proposé dans ISR utilise trois niveaux de représentation du monde. Les éléments de ces différents niveaux sont d'une part hiérarchiquement organisés et il existe d'autre part des relations "à plat" non hiérarchiques de type base de données relationnelle. La notion de "demon" ou de "procédure associée" à chaque "token" permet de gérer les traitements associés aux indices visuels.

### 3.1.2. Knowledge-Based Computer Vision [GOO 89]

*“Vision systems use a variety of complex data structures to represent their knowledge of images being processed and the application domain.”*

Goodman propose un nouveau langage de programmation et un nouveau système de gestion de données. Ces modèles reposent largement sur la technologie “orientée objet” (cf. figure 2.7.)



*figure 2.7. Relation entre objets dans un modèle du monde selon [GOO 89]*

### 3.1.3. Thin Line Code (TLC) [ JAG 89]

Dans TLC nous retrouvons sous une autre forme les différents niveaux de représentation évoqués dans ISR. La notion d’objet composite se prête bien à la représentation de ces différents niveaux allant du pixel à la sémantique de la scène. Cependant Jagadish fait une différenciation primordiale entre deux types de hiérarchie: physique et logique.

**Hiérarchie physique** : elle concerne la description de l’image indépendamment de toute considération sémantique concernant l’application. En d’autres termes cela signifie que la hiérarchie physique représente l’image en tant qu’image et non en tant que reflet du monde de l’application; dans TLC la hiérarchie physique se compose dans l’ordre depuis la base des “*pixels, chains, lines, composites, structures*”. Cette représentation est indépendante de toute

application et permet de créer des algorithmes stables sur des images venues de “mondes différents”.

**Hierarchie logique :** *“this is what the user wants to see”* et ce que pourrait être la fin d’une phase de reconnaissance sur l’image. Cette hiérarchie est spécifique à l’application et fait le lien entre le système de vision et ses utilisateurs pour une situation donnée.

La méthode utilisée consiste ensuite à créer des entités spécifiques qui servent de “ponts” entre la représentation physique de l’image et l’interprétation logique de celle-ci. TLC génère volontairement une séparation entre les données spécifiques au mode de description d’une image et celles spécifiques au mode de description du monde réel de l’application.

#### **3.1.4. Powervision [CON 89]**

“Les travaux de l’équipe de McConnel décrivent une architecture de système de vision basée sur quatre bases de données distinctes :

- PSDB (perceptual structure database), elle contient tous les objets d’une application donnée,
- EDB (environmental database), elle contient toutes les fonctions (traitements),
- LTDB (long terme database), elle contient une description des objets permanents d’un système de vision,
- PDB (programmers database), elle automatise le management des fonctions et des objets des autres bases.

Powervision est programmé dans un langage orienté objet, les objets sont directement stockés ainsi que leurs relations dans une base de données orientée objet.

McConnel fait clairement la distinction dans son système Powervision entre les données propres à une application particulière (perceptual structure database) et les données communes à toutes les applications de vision (long terme database), rejoignant en cela le concept de “hiérarchie physique et hiérarchie logique” évoqué par Jagadish [JAG 89] dans le système TLC.

McConnel apporte quelque-chose de nouveau par rapport aux autres systèmes de vision : il gère les “données traitement” comme des données classiques, et ce à travers “l’environmental database”. Ceci a pour effet immédiat d’améliorer l’ergonomie de son système et de réduire les temps de mise au point d’une nouvelle application. *“An underlying functional form built from an extendable set of macros that is used for expressing common processing operations for several different types of objects”*.

### 3.1.5. Base de données pour la représentation d'objets plats [QIN 88]

Hu Qing et Yang Jingyu ont élaboré un modèle de base de données appliqué à la reconnaissance d’objets plats partiellement cachés et déformés par perspective sur IBM PC/AT avec le système PCVISION. Ils décrivent une méthode de description du monde des objets plats à partir d’une approximation polygonale des contours. Cette description repose sur les critères suivants : concavité, direction globale des segments, longueur des segments, rotation angulaire. A partir de cette manière de voir le monde des objets plats, Qing introduit trois bases de données pour décrire de manière hiérarchique les objets plats:

- **Segment base** : le plus bas niveau dans la hiérarchie, elle contient l’ensemble des segments avec leurs attributs :
  - concavité, rotation, coordonnée du centre de gravité, longueur,
  - statistiques (nombre d’occurrences)
- **Boundary base** : le niveau intermédiaire, elle contient une description des contours en terme d’approximation polygonale, chaque segment d’un contour est à nouveau décrit avec sa position dans la séquence de segments. Il n’existe apparemment pas de lien entre la “boundary base” et la “segment base”.
- **Model base** : le niveau le plus élevé, elle contient la description complète des objets à identifier. Il existe cette fois un lien direct entre les contours décrits dans la “boundary base”, puisque ce sont ces même contours qui “forment” l’objet.

Cette mise en forme pourtant simple des données permet une application efficace d’algorithmes de reconnaissance dont la réalisation sera simplifiée par ce rangement. Les auteurs n’ont pas poussé plus loin leur étude et se sont limités à

la réalisation d'un modèle de données spécifique à la représentation des objets plats à partir d'une approximation polygonale. Les données concernant les traitements sont, comme le plus souvent, stockées de manière plus ou moins informelle dans les algorithmes utilisés. Quant aux données propres à une application particulière, elles n'apparaissent pas.

### **3.1.6. Base de données image pour une interprétation sémantique [PIZ 89]**

Arturo Pizaro décrit un modèle et un langage destiné à spécifier les contraintes de cohérence spatiale dans une base de données image. Il applique sa méthode à la compréhension de scènes urbaines avec des règles telles que : "deux voitures ne peuvent pas franchir un croisement au même moment" ou "un hôpital doit être situé à moins d'un kilomètre d'une école".

Pizaro définit une architecture à trois bases de données qui sont modélisées dans le formalisme entité association :

- **Constraint picture database** : elle renferme l'ensemble des règles de contraintes de l'application qui seront utilisées par l'algorithme d'interprétation.
- **Data dictionary** : elle contient l'ensemble des données visuelles (pixels, lignes ...).
- **Database** : elle contient une représentation du monde de l'application concernée. Les attributs visuels (pictorial primitive) des entités concernées font référence à des entités de la base "data dictionary" dans laquelle sont stockées toutes les informations visuelles, c'est ainsi que se forme le lien entre les deux bases (visuelle et réelle).

### **3.1.7. Conclusion sur l'analyse de différents systèmes de vision**

Le point commun de ces systèmes de vision est la volonté de représenter d'une manière structurée des informations présentes dans le processus d'interprétation. Chaque système ayant un but précis (ou un domaine d'application bien ciblé, mis à part Powervision [CON 89]). Cette représentation n'est jamais la même d'un système à l'autre; certains utilisent trois niveaux de représentation comme [BRO 89][QIN 88], d'autres distinguent des hiérarchies

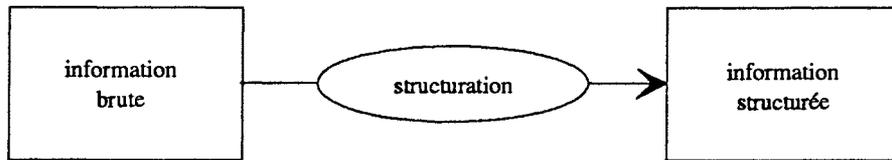
physiques et logiques [JAG 89]. A la vue des différentes réalisations, une certitude émerge : pour qu'un système de vision réduit à sa plus simple expression fonctionne, il lui faut au moins une modélisation interne des données spécifiques au procédé de vision [QIN 88]. Ces données visuelles sont selon les cas des descriptions de contour (lignes, arcs...) ou de régions (surfaces, périmètres...); viennent ensuite les systèmes plus évolués qui modélisent l'environnement de travail (pour des besoins d'interprétation) introduisant ainsi des données spécifiques à l'application [PIZ 89]; pour finir d'autres systèmes font une gestion des traitements effectués mais pour des raisons parfois différentes (soit pour améliorer l'environnement de travail, soit pour améliorer l'interprétation d'image).

### 3.2. Traitements et données

*"L'interprétation d'image est avant tout un traitement de l'information" :*

- **traitement** : on devrait plutôt dire "transformation" ou même "structuration",
- **information** : il y a l'information *avant* structuration (ou transformation) et il y a l'information *après* structuration, il doit y avoir au moins deux modes de représentation de l'information : avant et après.

Etymologiquement le mot "*informer*" signifie "*mettre en forme*" [LAB 87], l'information "après" a quelque chose en plus : une mise en forme, une structure. Ce phénomène était fortement et justement utilisé dans tous les systèmes de vision que nous avons présentés précédemment. Il est à l'origine des "*hiérarchies physiques et logiques*" [JAG 89], des "*long terme database*" et "*perceptual structure database*" [CON 89], des "*low level*" et "*high level*" [BRO 89] etc ..., c'est pourquoi nous avons également beaucoup de mal à nous fixer sur un "unique et idéal" mode de représentation, puisque nous avons au moins deux mondes à décrire, si ce n'est une infinité. Cette définition de l'interprétation d'image peut se schématiser simplement (cf. figure 2.8.).



*figure 2.8. Principe de base de compréhension*

Lorsque l'on associe des droites adjacentes à un objet dans telle ou telle position, on franchit un "*nouveau niveau d'organisation de l'information*" [LAB 87], à ce niveau une nouvelle logique intervient, celle du réel; on entre alors dans l'exercice de la compréhension de scène où des modèles de description du monde totalement différents interviennent ainsi que les moyens de les décrire (il s'agit souvent de systèmes à base de règles [PIZ 89]). **Si ces modèles n'existent pas, il est absolument impossible et illusoire de dépasser le stade de la reconnaissance d'objet.**

Le "traitement" est l'application permettant de passer d'un ensemble d'informations de niveau (n) à un ensemble d'informations de niveau (n+1), cette application fonctionne selon un "mode d'emploi" connu et appris. Ce "mode d'emploi" est une connaissance. Cette connaissance est stockée en pratique sous diverses formes : sous forme algorithmique dans les logiciels d'extraction de contour ou de droite, sous forme de modèles d'objet dans les logiciels de reconnaissance. Mais que se passe-t-il exactement lors de ce saut de niveau d'organisation? Voyons l'approche du biologiste Henri Laborit :

*"Communiquer, étymologiquement, signifie mettre en commun. Mais que met-on en commun? Une information. Ethymologiquement encore, une information est une mise en forme. Se pose alors la question de savoir ce qu'est la forme. La forme représente ce qu'un ensemble ajoute à la somme des éléments qui le constituent. Et ce qu'il ajoute, ce sont des relations. La mise en forme résulte donc des relations qui s'établissent dans un certain ordre entre les éléments d'un ensemble. On représente classiquement un ensemble par un cercle. A l'intérieur de ce cercle se trouvent des cercles plus petits qui représentent les éléments de cet ensemble. Ces éléments ne sont pas placés au hasard les uns par rapport aux autres. Ils établissent entre eux des relations. Wiener a insisté sur le fait que ces relations ne sont ni masse ni énergie; elles ne sont qu'information, alors que les éléments de l'ensemble seront toujours perçus comme masse ou*

*énergie. ... L'ensemble des relations constitue ce que l'on appelle la structure. Il constitue donc un nouvel ensemble, l'ensemble des relations, ..."* [LAB 87]. L'information est un ensemble de relations liant les éléments du niveau inférieur. Ces relations sont résumées et symbolisées au niveau supérieur par un élément. Par exemple : soit un ensemble de pixels (ces pixels sont des éléments de niveau (n) ), un certain nombre de ces pixels forment entre eux une droite, ils sont liés par cette relation d'appartenance à une droite, cette relation d'appartenance définit un ensemble dans l'ensemble des pixels, ce nouvel ensemble est symbolisé (ou baptisé en tant qu'ensemble) par un identificateur et est caractérisé par des critères globaux (propre à l'ensemble des pixels formant la droite) tels que l'orientation ou la longueur de la droite. Les traitements matérialisent et résument les liens qui peuvent exister entre deux niveaux d'informations successifs, ces liens peuvent physiquement se concrétiser dans une base de données sous la forme de pointeurs indiquant par exemple : la droite D1 et ses attributs (Longueur, Angle, Position) sont composés des pixels (x,y)<sub>1</sub> (x,y)<sub>2</sub> ...

Dans un article sur "l'information et l'entropie" paru dans l'Encyclopédia Universalis, Olivier Costa De Beauregard constate [BEA 89] : "*La cybernétique montre que l'acquisition de connaissance appartient à la sous-classe physiquement triviale des processus entropiques, retardés et causaux, tandis que l'action, ou le tri, appartient à la sous-classe des processus anti-entropiques, avancés, et finaux*" .

Le traitement d'image appartient à la catégorie du "tri d'information", c'est donc un procédé "anti-entropique", sa réalisation nécessite donc une grande quantité d'informations. **Le traitement d'image consomme plus de données qu'il n'en produit.** Ces données consommées sont incluses dans les algorithmes et dans les modèles utilisés par les systèmes de vision.

Il est de coutume de comparer un processus de compréhension d'image à un "entonnoir" qui, partant d'une information infinie (la scène réelle), va progressivement diminuer la quantité d'informations en se ramenant à une image de dimension finie (numérisation), puis au fil des traitements, à des informations de plus en plus qualitatives (identification).

Plus pragmatiquement, en vision les traitements se composent de quatre phases :

- la numérisation,
- les prétraitements
- la segmentation,
- l'interprétation.

### **3.2.1. La numérisation et les prétraitements**

La numérisation consiste à réaliser la transformation du signal analogique issu du capteur en un signal numérique exploitable par l'ordinateur. Cette transformation est électronique, elle s'effectue en temps réel. Dans le cas d'un système de vision matriciel, la numérisation forme une matrice de pixels. A chaque pixel est associée une (ou plusieurs pour la couleur RVB) grandeur caractéristique.

Les objectifs du prétraitement sont :

- la reconstruction d'une image "idéale" à partir de l'initiale; cette reconstruction va de la reconstruction géométrique (pour corriger les distorsions de l'optique), à la modification de l'échelle des niveaux de gris, de l'amélioration du contraste au lissage,
- la génération d'une image dérivée de l'image initiale (squelettisation, binarisation).

### **3.2.2. La segmentation et l'interprétation**

La segmentation d'une image consiste à créer tout d'abord des *ensembles de pixels connexes* (voisins) ayant des caractéristiques communes. En vision 2D ces ensembles peuvent être des contours ou des régions, en 3D des surfaces. Lorsque ces ensembles sont déterminés, on peut les caractériser à l'aide de primitives visuelles.

Il existe deux types de primitives visuelles :

- les *primitives globales*, calculées sur l'ensemble des pixels constituant le contour, la région ou la surface. On rencontre le plus souvent : périmètre, surface, excentricité, nombre d'Euler, compacité, moments, rectangle encadrant, barycentre, etc ...,
- les *primitives locales*, calculées sur une partie localisée des pixels constituant l'ensemble de départ (région). On rencontre le plus souvent : segment de droite, arc de cercle, angle, code de freeman, coin, etc ...

L'interprétation a pour but, à partir des primitives extraites de l'image lors de la segmentation, de répondre au problème posé initialement (quelle mesure? quelle position? quel objet?). Quelque soit le problème posé, le système de vision effectue toujours au moins une identification et une localisation (ce qui est l'équivalent du balayage sur les machines à mesurer), sauf dans les cas extrêmement simple tels que le test de présence. En effet, lorsque l'on veut mesurer une cote sur un objet A, il faut d'abord vérifier que c'est bien l'objet A que l'on mesure (à moins que l'on ait décidé d'en être sûr), et il faut trouver l'endroit de la cote (à moins que l'on ait décidé qu'elle soit toujours à la même position), donc il faut localiser l'objet A.

En conséquence, les principaux problèmes de la vision artificielle utilisée en milieu manufacturier se résument ainsi : identification, localisation.

Pour résoudre ces deux problèmes, plusieurs méthodes existent :

- les *méthodes discriminantes* (qui utilisent des primitives globales), parmi lesquelles nous distinguons les méthodes probabilistes (Bayes), déterministes (plus proche voisin),
- les *méthodes analytiques* (qui utilisent des primitives locales), parmi lesquelles nous distinguons les méthodes syntaxiques (grammaire, automate d'état finis) et les méthodes de prédiction-vérification.

D'autres techniques existent, difficilement classables :

- arbre décisionnels,
- recherche des cliques maximales, (graphes),
- relaxation, (corrélation).

Dans tous les cas, ces méthodes de traitement nécessitent pour leur fonctionnement un apport de connaissances; en effet, il faut comparer les primitives extraites d'une image à des primitives "connues d'avance" qui constituent les "modèles d'objets" du système de vision. Le format des modèles dépend des primitives et des méthodes employées.

La constitution de ces modèles constitue une étape clé de l'intégration d'un système de vision dans son environnement. Hélas, le plus souvent cette modélisation ne peut être réalisée autrement que par passage de l'objet à modéliser sous la caméra dans les conditions d'éclairage et de référence de l'application.

### 3.2.3. La représentation des traitements

Un traitement de vision est constitué d'un ensemble de composants de traitement qui sont : le prétraitement, la segmentation et l'interprétation. La constitution d'un programme de traitement d'image revient à fabriquer un arrangement de logiciels qui appartiennent à ces trois catégories.

Nous avons donc déterminé l'*individu* MONTAGE TRAITEMENT (cf. figure 2.9.). Le MCD de référence permet de classer les logiciels de vision disponibles par rapport aux données qu'ils consomment et produisent.

Les *occurrences* des *individus* TRAITEMENT, SEGMENTATION et PRETRAITEMENT sont des programmes source types, des bibliothèques, des fonctions etc ... Ces occurrences sont les matériels logiciels de base (à l'image de l'Environmental DataBase qui contient tous les traitements types utilisés dans le système PowerVision de [CON 89]) nécessaires à la réalisation de programmes de traitement visionique.

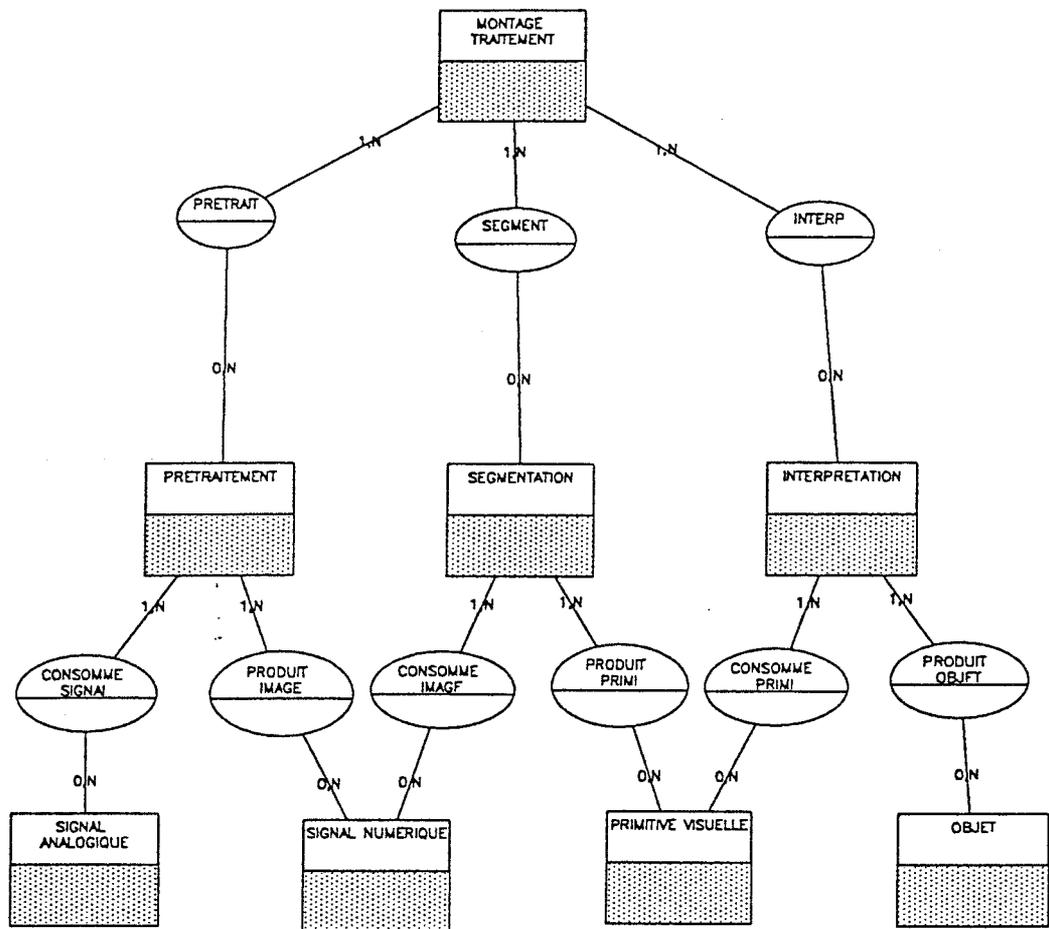


figure 2.9. Extrait du MCD de référence de vision : *représentation des traitements*

### 3.2.4. La représentation du monde visuel

Le système de vision doit disposer d'une représentation interne du monde visuel (cf. figure 2.10a. et 2.10b.). Comme dans le système ISR de [BRO 89], nous distinguons trois niveaux de représentation : le niveau "image", le niveau "primitive visuelle" et le niveau "réel". Le niveau "image" modélise l'ensemble des données issues de la prise d'image et des prétraitements; le niveau "primitive visuelle" modélise les données fabriquées par la segmentation; le niveau "réel" représente les données issues de l'interprétation des primitives. Le réel (*individu OBJET*) ne peut être décrit que par rapport à l'application; dans notre cas d'intégration dans une cellule flexible, le réel est constitué des gammes et des

entités de fabrication pour les contrôles de qualité sur le produit. mais aussi des positions/localisations pour des contrôles de manutention.

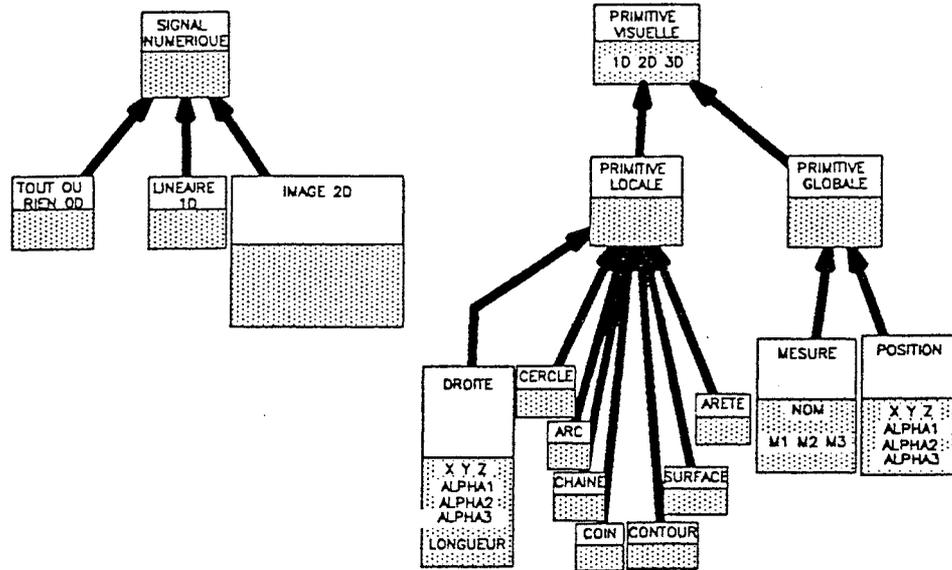


figure 2.10a. Extrait du MCD de référence de vision : représentation du monde visuel

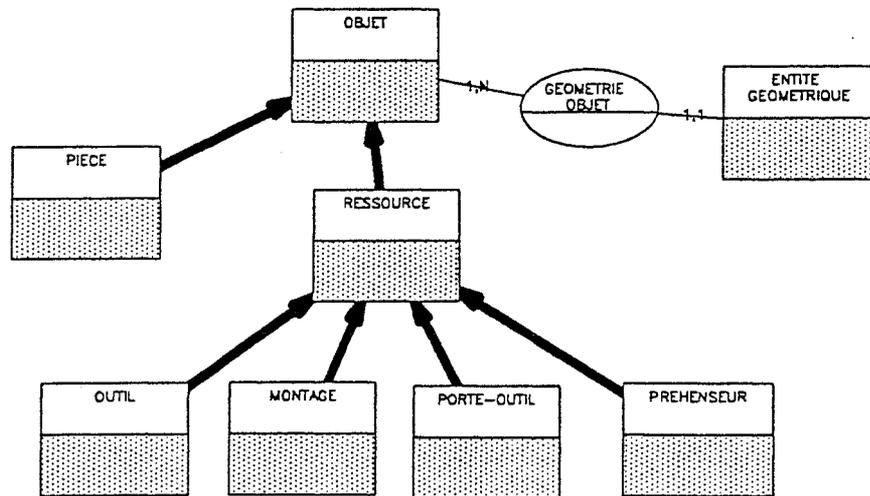


figure 2.10b. Extrait du MCD de référence de vision : représentation du monde réel dans la cellule

### 3.2.5. Les programmes et les données utilisés par les traitements

L'exécution d'une opération de vision nécessite le stockage à la fois des programmes de vision et des connaissances utiles à leur fonctionnement.

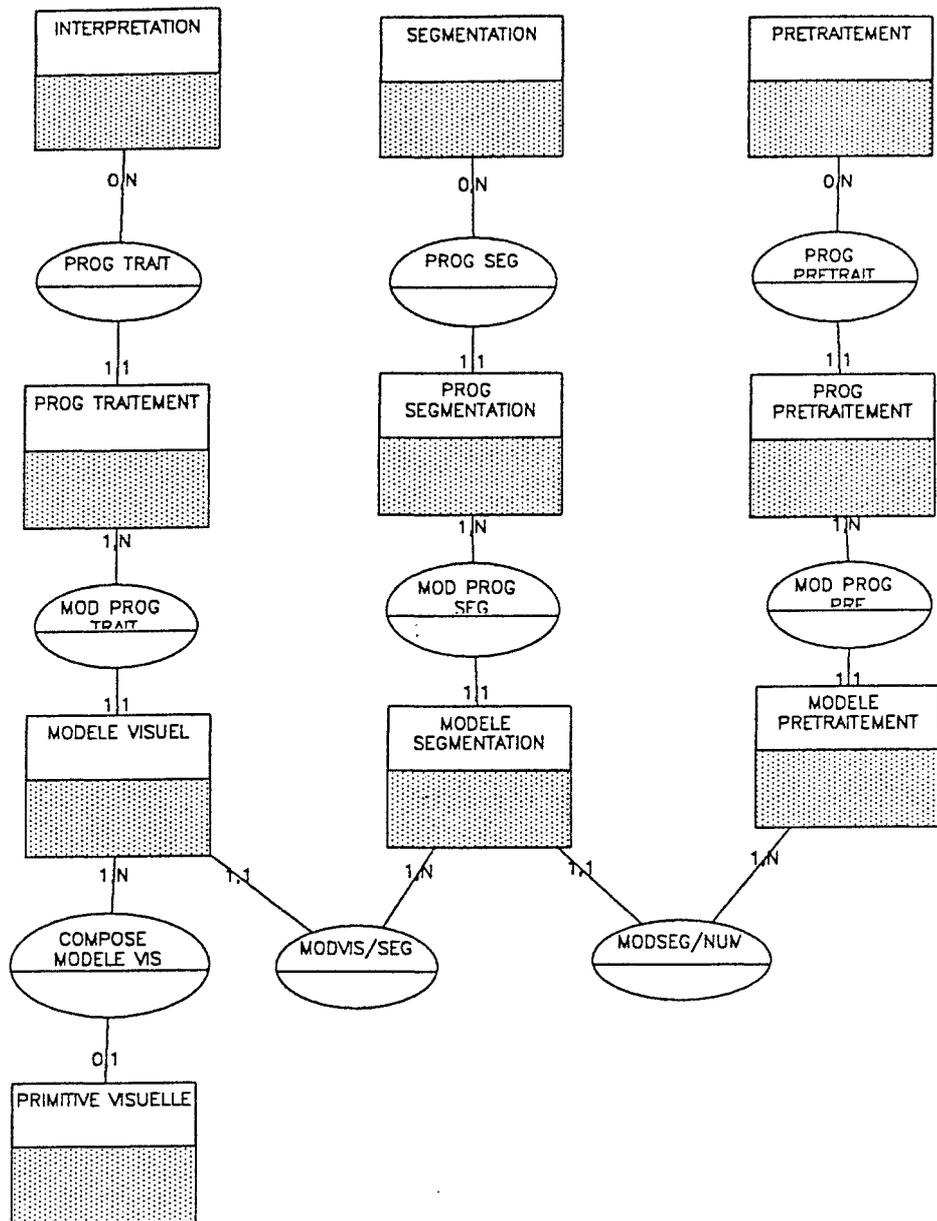


figure 2.11. Extrait du MCD de référence de vision : les exécutable et les données utilisés par les traitements

Dans le MCD de référence de la figure 2.11., les *occurrences* des *individus* PROG TRAITEMENT, PROG SEGMENTATION , PROG PRETRAITEMENT sont des programmes exécutables de vision.

Les logiciels d'interprétation utilisent des modèles d'objet, des arbres de stratégie, etc... Nous avons regroupé ces notions dans l'*individu* MODELE VISUEL qui est relié à l'*individu* PRIMITIVE VISUELLE.

Les logiciels de segmentation ont une partie de leur connaissance codée "en eux" et une partie externe : les paramétrages (taille maximum de droite à extraire, étalonnage, type de primitives, etc ...). Nous avons regroupé ces données dans l'*individu* MODELE SEGMENTATION.

Il en est de même pour l'exécution des prétraitements qui doivent avant chaque utilisation être configurés (en mode binaire, contour, etc...). Nous avons regroupé ces données dans l'*individu* MODELE PRETRAITEMENT.

A ce niveau d'abstraction (conceptuel de référence), nous ne pouvons pas préjuger du mode de représentation et de stockage de ces données. Dans un MCD cible d'une application particulière, il est possible que certains *individus* comme PROG SEGMENTATION ou PROG PRETRAITEMENT disparaissent au profit d'autres *individus* (PROG TRAITEMENT) dans le cas où les traitements sont rassemblés dans un seul exécutable.

### 3.2.6. Les données issues du traitement

Chaque composante de traitement "transforme" des informations de niveau (n) en des informations de niveau (n+1). Ces niveaux sont : le signal numérique (image binaire), la primitive visuelle (droite), l'objet (pièce)(cf. figure 2.12.).

Le choix de l'exécution d'un programme de traitement (une *occurrence* de l'*individu* PROG TRAITEMENT) entraîne la sélection d'un ensemble non vide d'*individus* MODELE VISUEL, chaque *individu* MODELE VISUEL est relié à un seul *individu* MODELE SEGMENTATION, chaque *individu* MODELE SEGMENTATION est relié à un seul *individu* MODELE PRETRAITEMENT avec leur programme associé. Leur exécution fabrique (ou ne fabrique pas) successivement des *occurrences* des *individus* SIGNAL CAPTE (image binaire),

PRIMITIVE VUE (droite) et OBJET VU (pièce). Ces *occurrences* sont renseignées par les *individus* de représentation du monde visuel.

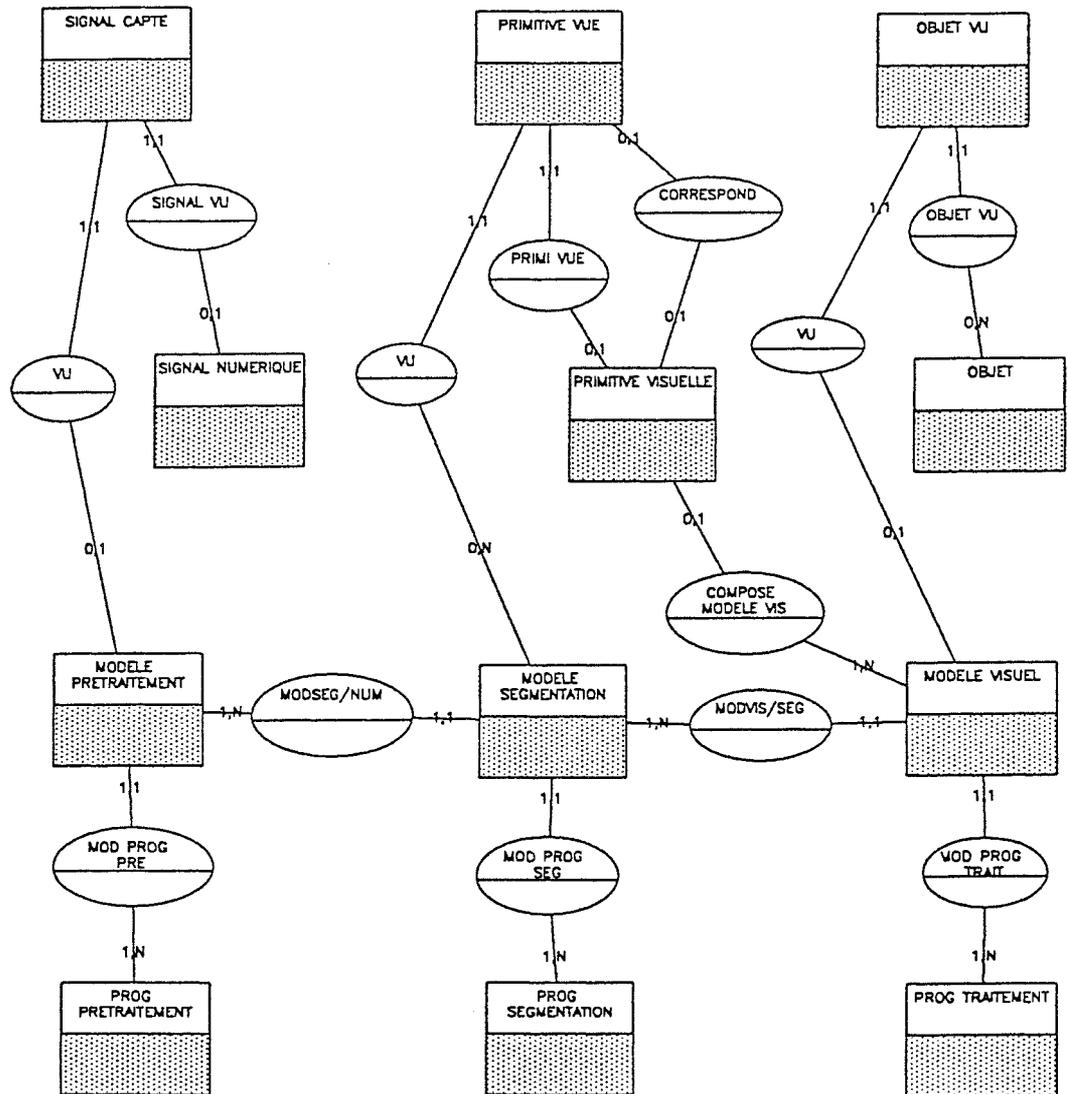


figure 2.12. Extrait du MCD de référence de vision : les résultats d'une opération de vision

#### 4. L'individu SYSTEME DE VISION

En résumé nous pouvons définir l'*individu* SYSTEME DE VISION qui est l'ensemble des différentes parties physiques et traitements que nous avons étudiées ultérieurement (cf. figure 2.13.).

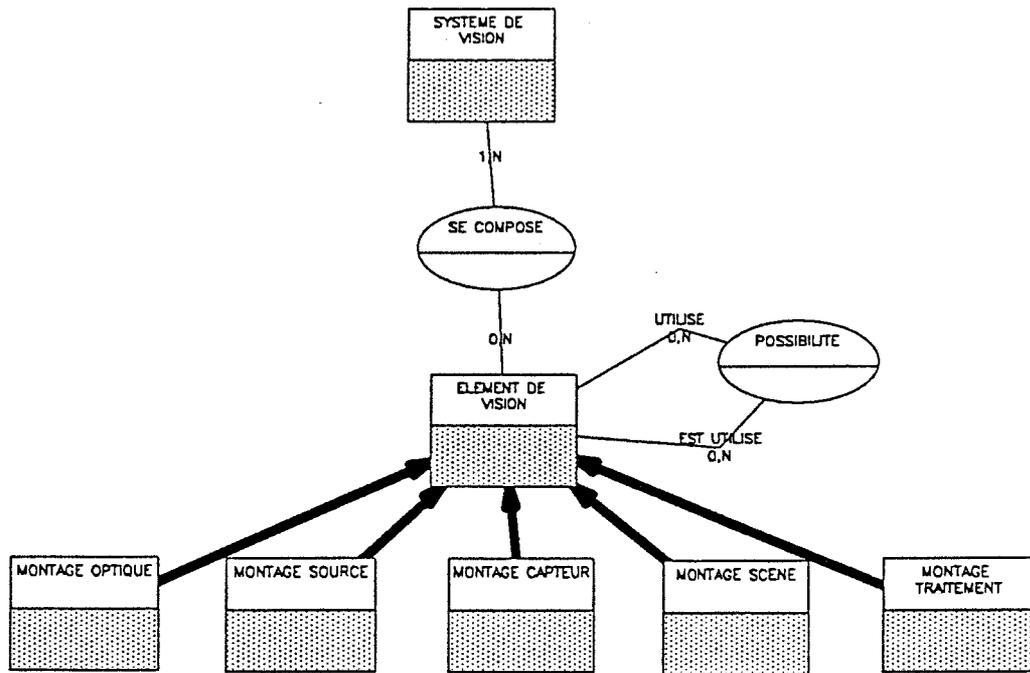


figure 2.13. Extrait du MCD de référence de vision : *Le système de vision*

#### 5. Application : système de vision matricielle

Nous présentons le système de vision dont nous nous sommes servi dans la maquette réalisée dans le cadre du contrat de recherche MRT : "Auto-contrôle de cellule flexible d'usinage, intégration dans un ensemble CFAO". Dans ce paragraphe nous nous limitons à faire une description physique de notre système de vision

### **5.1. Les optiques**

Voici nos composants optiques :

- objectif vidéo grand angle 6 mm, f 1.4,
- objectif photographique macro Nikor 50 mm, f 2.8,
- objectif spécialisé micro grossissement x 40,
- objectif vidéo 30 mm, f 1.8,
- adaptateur photo video F C NIKON,
- bagues allonge montures C et F.

### **5.2. Les capteurs**

Nous disposons :

- d'une caméra vidicon SONY,
- d'une caméra miniaturisée SYSTEME SUD,
- d'un trépied pour appareil photographique,
- d'un statif,
- d'une carte de numérisation EDGE 90, 4 entrées vidéo, 256 x 464 pixels, avec binarisation et extraction de contour en temps réel.

### **5.3. Les éclairages**

Nous disposons :

- de quatre projecteurs halogènes,
- de trépieds télescopiques avec articulation.

### **5.4. Les scènes**

Nous avons réalisé plusieurs applications avec le même système de vision, en

conséquence nous avons plusieurs scènes :

- test de présence de l'objet de référence NASA (cf. figure 2.14.) sur poste de brut, avec la caméra sony montée à la verticale sur un trépied,
- identification et localisation de l'objet NASA sur le poste de brut, avec la caméra sony montée à la verticale sur un trépied, le fond du poste de brut est noir,

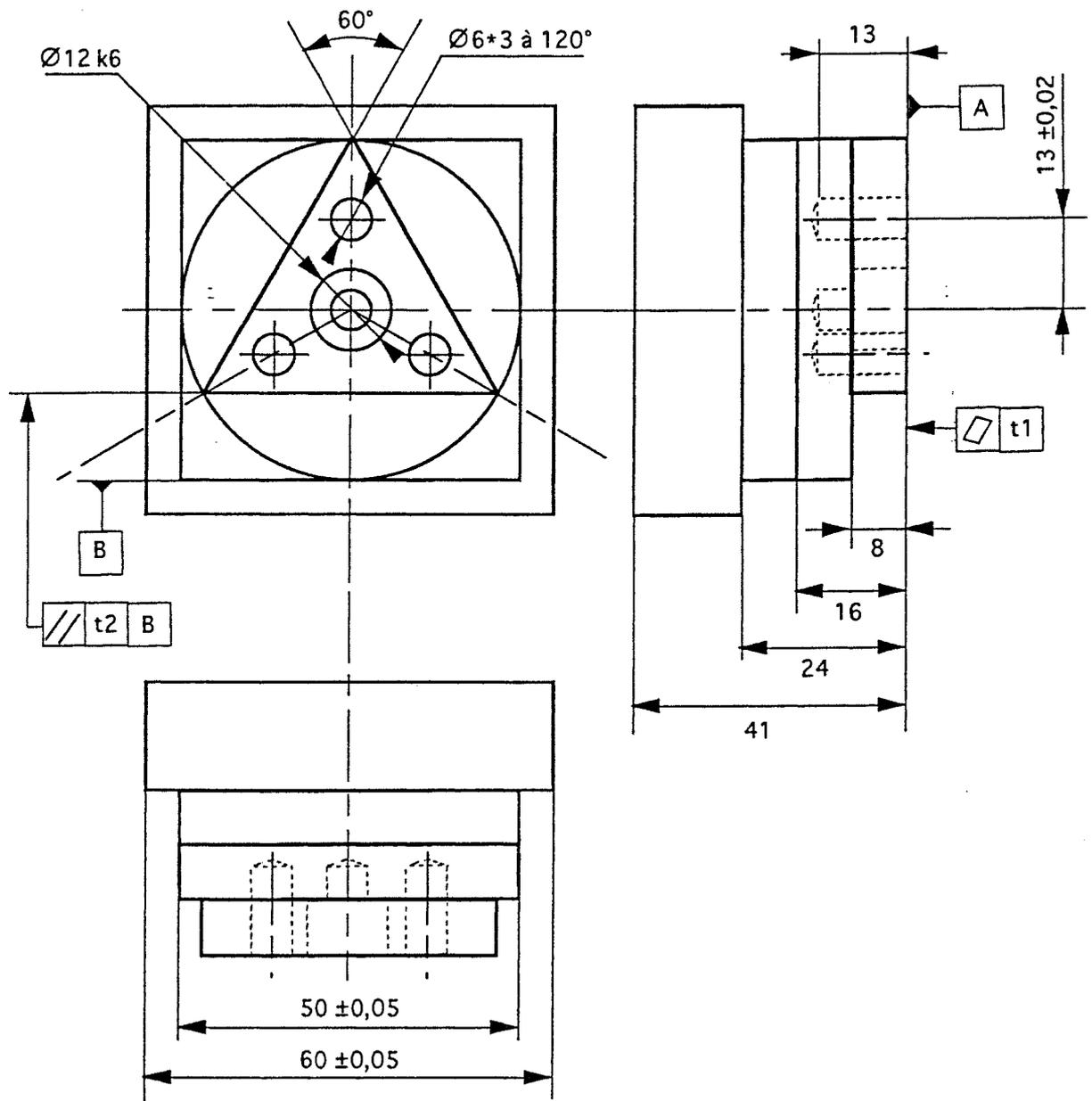


Figure 2.14. La pièce NASA

- test de fermeture/ouverture sur porte-pièce, caméra embarquée verticalement sur l'axe Z du CHARLY-ROBOT,
- test de présence de l'objet NASA sur porte-pièce dans le CHARLY-ROBOT, caméra embarquée verticalement sur l'axe Z, le fond du porte pièce a du être peint en noir,
- contrôle de présence et de dimension d'une entité de perçage sur la pièce NASA montée dans son porte-pièce dans le CHARLY-ROBOT, la caméra est embarquée sur l'axe Z.

## 5.5. Les traitements

### 5.5.1. Les prétraitements

Les prétraitements sont entièrement réalisés par la carte EDGE 90, il suffit de configurer la carte en chargeant un **fichier de configuration** et d'effectuer la prise d'image (cf. figure 2.15.). Lorsque le PC de vision s'initialise, un **fichier de commandes** du système est chargé. Ce fichier contient la description des différentes opérations (prévues d'avance) que peut réaliser le capteur. Par exemple : l'opération 3 correspond à la prise de l'image nommée "h:brut.vis" avec la carte configurée selon le fichier "brut.cfg". L'image est transférée de manière transparente par le réseau ETHERNET.

<pre> /*nom_de_l'operation*/ MESURE_TROU1 /*fichier_image*/ v:trou1 /*fichier_de_config_de_EDGE_90*/ trou1.cfg /*numero_de_camera*/ 2 /*employeur*/ STATION_USINAGE </pre>	<pre> CHAMP_CAMERA=1 MODE_TRAVAIL=1 MODE_VISU=2 NUM_LUT=0 SEUIL_BINAIRE=21 NUM_CAMERA=2 </pre>
--	--

*figure 2.15. Fichier de commande et fichier de configuration de la carte EDGE : contour binaire*

Lorsque nous effectuons un test de présence, nous utilisons une image binarisée; quand nous réalisons une identification ou une localisation, nous configurons la carte en mode contour binaire.

### 5.5.2. La segmentation

Pour le test simple de présence, la segmentation de l'image consiste en un simple comptage de pixels. Le résultat de la segmentation est un nombre de pixels (primitive globale).

Dans les autres cas (identification, localisation, mesure), la segmentation est réalisée par la couche intermédiaire de logiciel CAIMAN, développée par ITMI en langage fortran. Le logiciel CAIMAN fournit un suivi de contour avec calcul en parallèle de primitives globales sur contours fermés et de primitives locales sur contours quelconques. Nous nous servons essentiellement des primitives "droite" et "cercle". Nous utilisons un **fichier d'étalonnage** (cf. figure 2.16.) pour paramétrer la segmentation, le reste des informations de configuration est stocké dans le programme lui même; nous réécrivons un nouveau programme à chaque nouvelle application.

```
fichier etalonnage brut.eta
x/y  0.357724
x    0.458333
y    1.261 250
```

*figure 2.16. Fichier d'étalonnage*

### 5.5.3. L'interprétation

Dans le cas de test de présence, l'interprétation consiste en une comparaison du nombre de pixels trouvés dans l'image à un seuil. Ce système est très fiable et très rapide, mais il n'identifie pas l'objet présent.

Dans le cas d'une identification ou d'une localisation d'objet en vrac planaire, nous utilisons un algorithme de prédiction-vérification [AYA 83] [SOU 83] [ROC 88]. Cet algorithme utilise une segmentation à base de droite, les **fichiers modèles d'objet** contiennent une description d'objet visuel à l'aide de primitives visuelles de type droite (cf. figure 2.17.).

```

modele visuel      nasa
objet ressemblant  0
symetrie           4.000000
nombre de droites  4
contour            240.000000
droite passe n°1   15.000000
seuil1             2.000000
seuil2             0.300000
seuil3             4.000000
seuil4             4.000000
delta angle        0.200000
delta distance     5.000000

droite 1
longueur           60.000000
angle              1.570796
sens               0
position x         0.000000
position y         0.000000

droite 2
longueur           60.000000
angle              0.000000
sens               0
position x         30.000000
position y         30.000000

droite 3
longueur           60.000000
angle              0.000000
sens               0
position x         30.000000
position y         -30.000000

droite 4
longueur           60.000000
angle              1.570796
sens               0
position x         60.000000
position y         0.000000

```

*figure 2.17. Fichier modèle d'objet pour la prédiction vérification*

Dans le cas de la mesure du diamètre d'un perçage sur la pièce NASA, l'interprétation consiste à vérifier qu'un trou a bien été trouvé là ou on avait prévu dans l'image, puis d'enregistrer le diamètre du trou trouvé dans la base de données s'il est présent. Ce système utilise un **fichier modèle d'objet** correspondant au modèle de l'image prévue, après segmentation (cf. figure 2.18.).

```
modele visuel trou1
nb primitive 1

cercle 1
diametre 3.140305
position x 11.290605
position y 9.359888
```

*figure 2.18. Fichier modèle d'objet pour la mesure d'un perçage*

## **5.6. Conception du Système d'Information intégrant le contrôle qualité par vision**

Cette application traite le cas précis d'un contrôle d'une pièce dans la cellule flexible d'usinage. Nous allons montrer quelle doit être l'utilisation des MCD de Référence dans la démarche de conception du Système d'Information de la cellule. Nous cherchons à concevoir la partie du SI destinée à permettre le stockage des résultats d'une opération de contrôle par vision sur le produit.

### **5.6.1. MCD de Référence**

Le MCD de Référence qui concerne cette application est celui qui modélise les résultats d'une opération de vision (cf. figure 2.12.) que nous avons étudié au paragraphe 3.2.6. de ce chapitre.

### **5.6.2. MCD partiel**

Nousinstancions le MCD de Référence à notre système de vision particulier, nous obtenons ainsi un MCD partiel (cf. figure 2.19.). Ce modèle ne préjuge pas

encore du type de contrôle effectué par notre système de vision ni du lieu de stockage des données. Il pourrait être réutilisé pour un autre type de contrôle (test de présence, localisation). Par rapport au MCD de Référence, le MCD Partiel contient moins d'individus. En effet notre système de vision réunit en un seul exécutable l'ensemble "segmentation et interprétation"; d'autre part, les primitives visuelles qui sont extraites de l'image ne sont pas accessibles en dehors du programme de traitement.

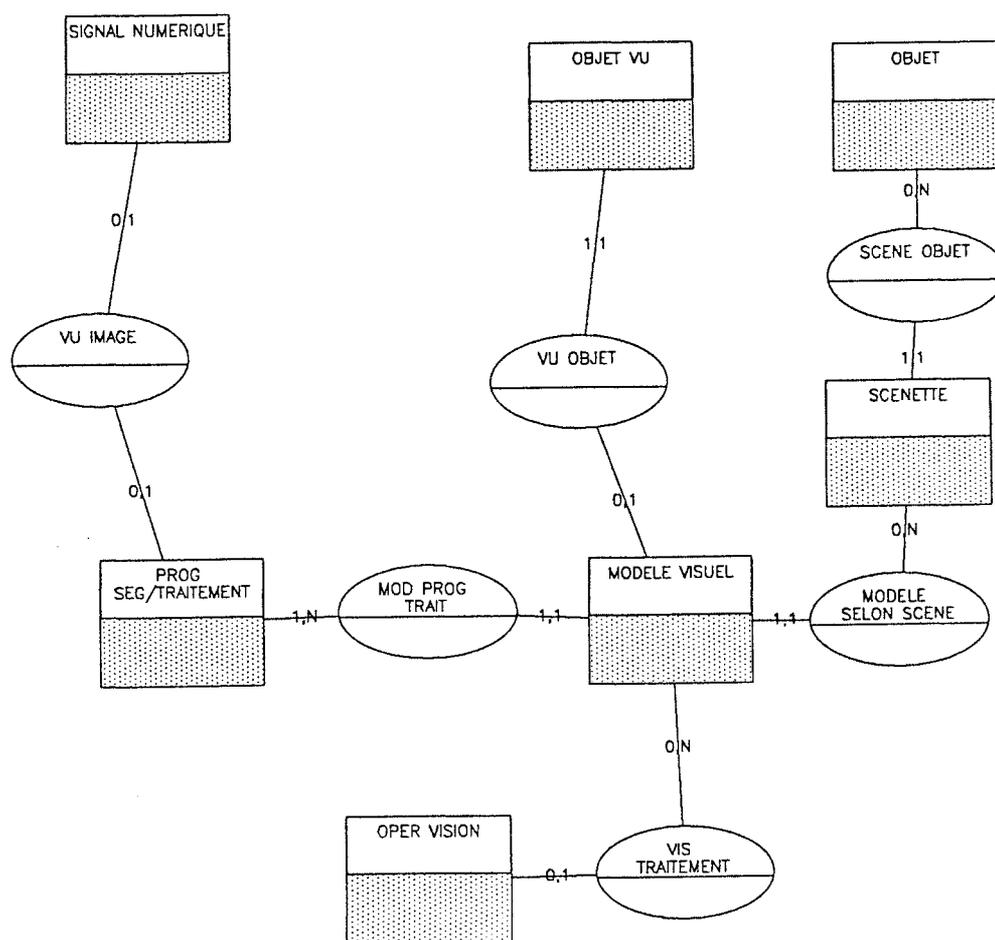
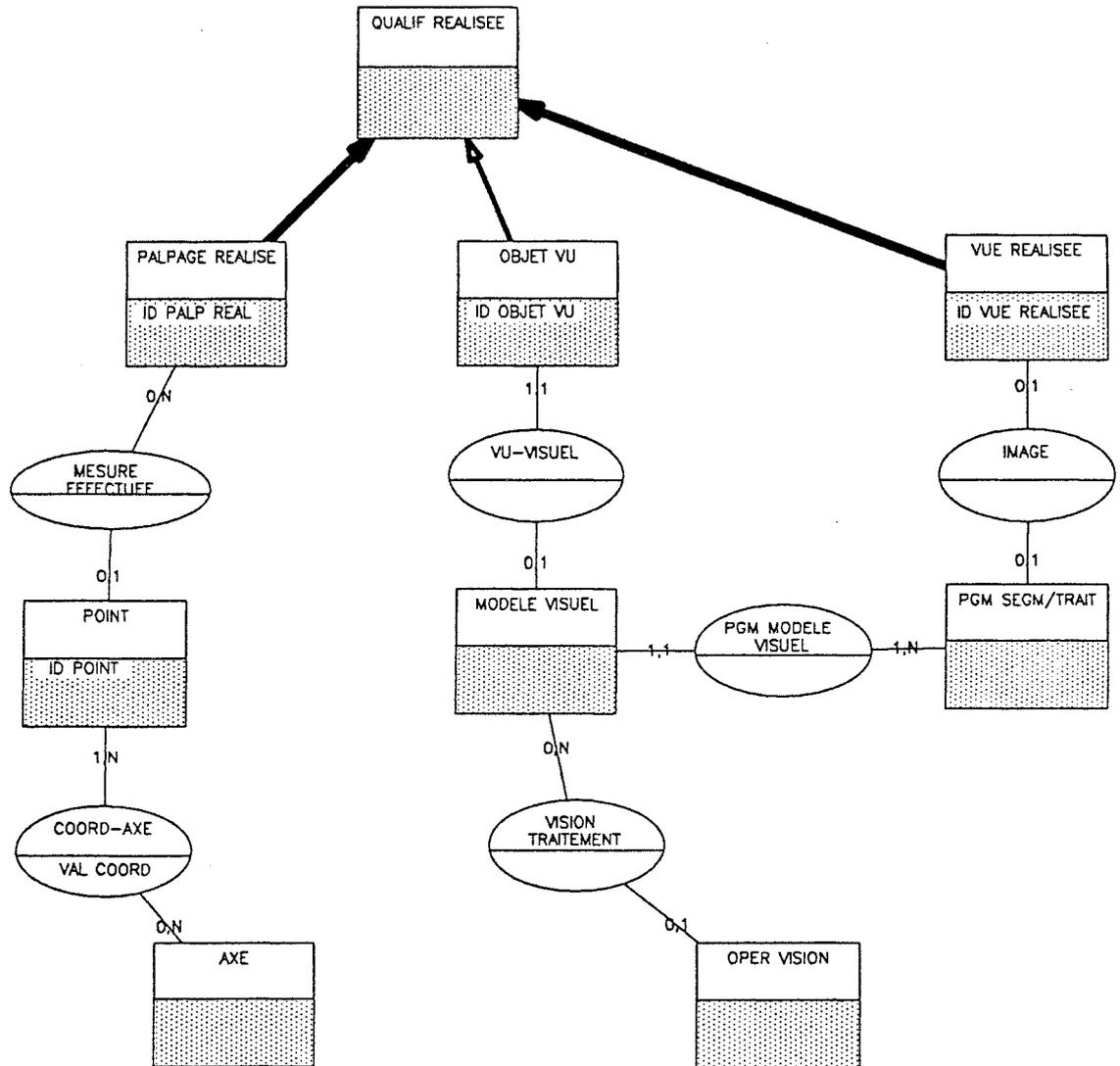


figure 2.19. MCD Partiel : les résultats d'une opération de vision

### 5.6.3. MOD particulier



*figure 2.20. MOD Particulier : les résultats d'une opération de vision dans la base "pilote et qualité"*

Le passage du MCD partiel au MOD particulier (cf. figure 2.20.) prend en compte la répartition des données sur un site particulier et le type de contrôle effectué. Il n'est pas nécessaire de représenter le MCD particulier. Dans notre application, les résultats de la mesure sont stockés dans la base de données

"qualité et pilotage" [COU 91]. Notre MOD est donc directement intégré au MOD de cette base lors de la conception de celle ci. Nous aurions pu choisir de répartir autrement nos données "résultats de contrôle" pour améliorer les temps de réponse du système, notamment en choisissant comme lieu de stockage la station ou le VMD supporté par la couche MMS du réseaux MAP. Dans ce cas le MOD particulier aurait été différent ainsi que la réalisation physique. Mais les niveaux conceptuels demeurent identiques.

#### 5.6.4. MLD particulier

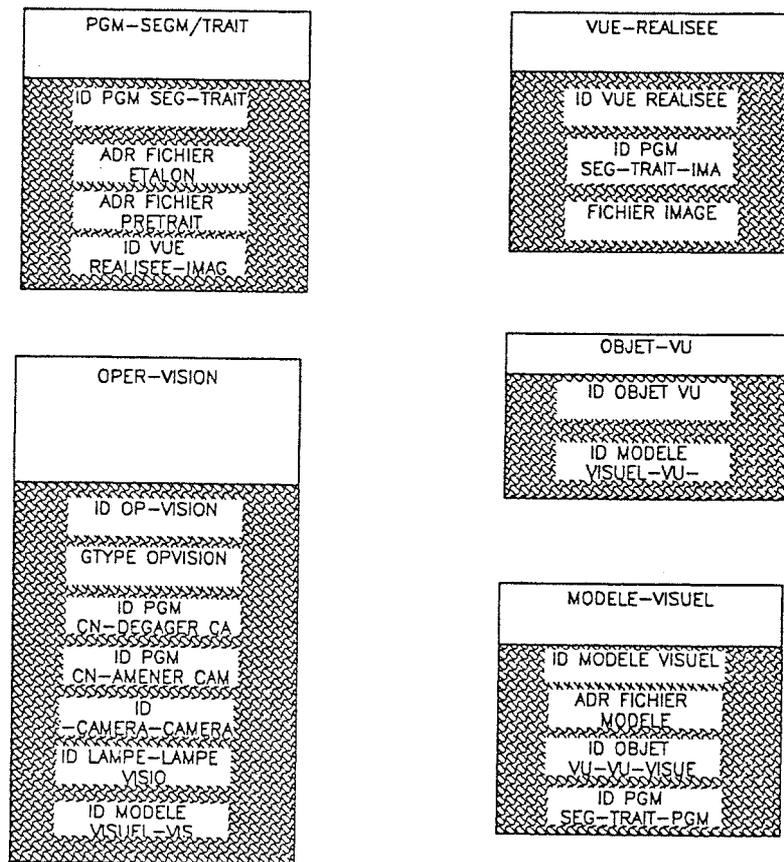


figure 2.21. MLD Particulier : les résultats d'une opération de vision dans la base "pilotage et qualité"

Le MLD particulier est directement dérivable du MOD particulier (cf. figure 2.21.). Il est décrit en formalisme relationnel

### 5.6.5. Génération du code SQL

L'atelier logiciel MEGA permet la génération automatique du code (cf. figure 2.22.) en langage SQL pour la génération des tables de la base de données relationnelle SQL SERVER. Les programmes de traitement de vision doivent être augmentés d'une interface SQL pour permettre la mise à jour automatique de la base lors de leur exécution. Des requêtes SQL sont ajoutées à la fin des programmes de traitement de vision (cf. figure 2.23.)

```
create table axe
    (nom_axe varchar(1) not null)
go
grant delete, insert, select, update
on axe
to ac_cfu
go
create table coord_axe
    (val_coord float null,
    id_pgm_seg_trait_ima smallint null,
    id_point smallint not null,
    nom_axe varchar(1) not null)
go
grant delete, insert, select, update
on coord_axe
to ac_cfu
go
create table modele_visuel
    (adr_fichier_modele varchar(10) null,
    id_modele_visuel smallint not null,
    id_pgm_seg_trait_pgm smallint null,
    id_objet_vu_vu_visue smallint null)
go
grant delete, insert, select, update
on modele_visuel
to ac_cfu
go
create table objet_vu
    (id_objet_vu smallint not null,
    id_modele_visuel smallint null)
go
grant delete, insert, select, update
on objet_vu
to ac_cfu
go
create table oper_vision
    (id_op_vision smallint not null,
    gtype_op_vision varchar(20) null,
    id_t_camera_camera_p smallint null,
    id_pgm_cn_amener_cam smallint null,
    id_pgm_cn_degager_ca smallint null,
    id_lampe_lampe_visio smallint null,
    id_modele_visuel_vis smallint null)
go
grant delete, insert, select, update
on oper_vision
to ac_cfu
go
create table palpage_realise
    (gtype_palp_real varchar(20) null,
    id_palp_real smallint not null)
```

*figure 2.22. Génération du code SQL à partir du MLD*

```

/* initialisation de SQL SERVER ETHERNET */
dberrhandle(err_handler);
dbmsghandle(msg_handler);
login = dblogin();
DBSETLUSER(login, "rochette");
DBSETLPWD(login, "");
DBSETLAPP(login, "vision");
if( (dbproc = dbopen (login, "SERVEUR")) == NULL)
printf ("\nSQLServer : erreur dbopen \n");
/* initialisation caiman */
congbl(); inivog();
/* recuperation des parametres */
operation=atoi(argv[1]);
strcpy(etalon,argv[3]);
strcpy(image,argv[2]);
for (i=0;i<argc-5;i++) strcpy(objet[i],argv[i+5]);
/* scene image */
strcpy(etat_cam,image);
/* chargement etalonnage */
eta_restor(etalon);
/* validation perimetre cercle barycentre */
indice=256; valglo(&indice);
/* chargement du modele */
for (i=0;i<argc-5;i++) lire_mod(objet[i]);
/* affichages graphiques */
syst.dbg=32; visu=0;
/* analyse de l'image */
locacer(model[0].nom);
/* ecriture des resultats */
for (i=0;i<model[mod].nbpri;i++)
if (trouv[i]!=10000)
{
diam=drte.ind[trouv[i]].mesure[0];
x=drte.ind[trouv[i]].loca.pt_parti.x;
y=drte.ind[trouv[i]].loca.pt_parti.y;
/* ecriture des resultats dans la base SQL */
ret=cr_oper_vision(operation,image);
entite=ajoute_ent_mesuree("perchage");
ret=ajoute_param_mesure(operation,"rayon",diam*eta[i]);
ret=ajoute_param_mesure(operation,"x",erreurx[i] );
ret=ajoute_param_mesure(operation,"y",erreury[i]);
}
else
{
ret=ajoute_param_mesure(operation,"defaut",0.);
}
cr=1;
/* Termine la session avec SQL Server */
dbexit();
/* compte rendu OS2 */
DosExit(1,cr);

```

*figure 2.23. Requêtes SQL dans un programme vision*

## Chapitre 3

# Intégration dans le système physique de production

1. Problématique .....	1
2. La cellule flexible .....	2
2.1. modèle de référence des processus opératoires .....	2
2.2. Le niveau cellule.....	3
2.3. Le niveau station.....	5
3. Le rôle d'un système de vision dans la cellule.....	5
3.1. La vision comme poste de contrôle de la pièce dans la cellule .....	6
3.2. La vision comme outil de contrôle de la pièce et des processus opératoires sur une machine .....	6
3.3. La vision "réflexe" .....	7
4. Objectifs pour la vision .....	7
4.1. Un capteur flexible, multi-fonctions, multi-utilisateurs .....	7
4.2. Un capteur intégré.....	8
4.2.1. Aspect communication .....	8
4.2.2. Aspect contrôle commande.....	8
4.3. Un capteur auto-contrôlé.....	8
4.4. Un capteur ouvert .....	9
4.5. En bref : les yeux de la cellule .....	10
5. Décomposition fonctionnelle.....	11
5.1. Fonction préparer triplet "source-objet-recepteur" .....	11
5.2. Fonction capter information .....	11
5.3. Fonction traiter l'information .....	13
5.4. Fonction contrôler l'opération de vision .....	13
5.5. Les données nécessaires à l'exécution d'une opération de vision.....	13
6. Les objets "intelligents" en automatique .....	14
6.1. Le concept de filtre .....	15
6.2. Le module fonctionnel d'automatisme.....	16
6.3. L'actionneur "intelligent" .....	17
6.4. Le capteur "intelligent" .....	18
7. Application à l'intégration physique d'un système de vision dans une cellule flexible d'usinage .....	18
7.1. L'architecture physique de la cellule .....	18
7.1.1. L'ordinateur cellule.....	19
7.1.2. La station du CHARLY-ROBOT .....	19
7.1.3. La station POSTE DE BRUT .....	21
7.1.4. Les équipements partagés .....	21
7.2. Les contrôles réalisés .....	21
7.2.1. Les contrôles sur le POSTE DE BRUT .....	21
7.2.2. Les contrôles sur le poste d'usinage CHARLY-ROBOT.....	22

7.3. Exemple : contrôle d'un perçage sur fraiseuse CHARLY-ROBOT .....	22
7.3.1. Fonction "actionner lampe" .....	26
7.3.1.1. Prototypage .....	26
7.3.1.2. Réalisation .....	26
7.3.2. Fonction "positionner caméra" .....	28
7.3.3. Fonction "prendre image + prétraiter + transférer" .....	29
7.3.3.1. Prototypage .....	29
7.3.3.2. Réalisation .....	29
7.3.4. Fonction "segmenter + interpréter image" .....	30
7.3.5. Fonction "contrôler l'opération de vision" .....	31
7.3.5.1. Les traitements de contrôle de l'opération de vision.....	31
7.3.5.2. Les données de contrôle de l'opération de vision .....	32
7.3.5.2.1. MCD de Référence .....	32
7.3.5.2.2. MCD partiel.....	32
7.3.5.2.3. MOD particulier.....	32
7.3.5.2.4. MLD particulier et code SQL.....	34

## Chapitre 3

# Intégration dans le système physique de production

### 1. Problématique

Si l'expert de vision est tout à fait à l'aise dans la mise au point d'une optique particulière ou d'un traitement de vision, s'il lui est plus ou moins facile de par son savoir-faire en vision de choisir un algorithme adapté à une situation donnée (travailler sur les contours des objets, sur des régions, utiliser tel ou tel prétraitement), la situation peut lui échapper quand il s'agit d'installer le système de vision "déjà conçu" dans le site industriel (nous nous sommes fixé comme cadre de référence physique un environnement de cellule flexible).

Une multitude de problèmes se pose alors : comment faire communiquer le système de vision avec son environnement? Comment assurer une certaine qualité aux informations fournies par le système? Comment surveiller le système? Comment transmettre les résultats? Quelles données le système doit échanger? Qui peut utiliser le système? Comment allumer les lampes? Quelles architectures doivent avoir les différents équipements? Qui commande les équipements?

L'objectif de ce chapitre est de parvenir à définir une "architecture physique et fonctionnelle de référence" d'un système de vision intégré dans une cellule flexible. Cette architecture sera modélisée en formalisme SADT. Nous validerons notre modèle de référence par son instanciation à un cas particulier dans la cellule de la maquette du projet MRT.

## 2. La cellule flexible

Nous nous plaçons dans le cadre du modèle de référence de système de production : N.I.S.T. [CIM 90] (cf.figure 1.1. chap I).

Ce modèle hiérarchique décompose le système de production en cinq niveaux : usine, atelier, cellule, station, machine. Ce modèle nous sert de base pour la définition d'une cellule et des différents niveaux qui la compose.

L'intégration de machines autonomes en une cellule revient à proposer une architecture permettant :

- une reconfiguration rapide de la cellule, en cas de défaillance d'un équipement ou en vue d'une nouvelle fabrication,
- de faire communiquer et exploiter des matériels existants,
- de décentraliser "l'intelligence", augmenter l'auto-contrôle des équipements,
- d'intégrer le contrôle-qualité en ligne.

Les notions d'intelligence répartie et d'autonomie sont à la base de la conception d'un pilotage de la cellule à deux niveaux : le niveau cellule et le niveau machine [ADI 86] [VOG 86].

### 2.1. modèle de référence des processus opératoires

Avant de définir le rôle de chacun de ces niveaux, il est nécessaire de préciser la notion de processus de fabrication. La cellule flexible est chargée de réaliser la fabrication d'un produit selon un certain processus opératoire. Notre modèle de référence des processus opératoires est le suivant :

*Définition d'une gamme* : une gamme est un ensemble de sous-phases qui s'exécutent dans la cellule.

*Définition d'une sous-phase* : une sous-phase est un ensemble d'opérations qui s'exécutent sans démontage de la pièce. Un ensemble de sous-phases réalisées sur une même machine forme une *phase*.

*Définition d'une opération* : une opération est un ensemble d'actions élémentaires concourant à la réalisation d'une transformation sans changement d'outil; on distingue les opérations d'usinage (perçage, dressage, etc...), les opérations de contrôle (palpage, contrôle par vision, etc...), les opérations de manutention (serrage, desserrage d'un porte pièce, etc..).

*Définition d'une action* : une action est une activité élémentaire réalisée par un équipement de base d'une machine ou par un péri-équipement (ouverture d'un sas, allumage d'une lampe, déplacement d'un axe, indexation d'une tourelle, mise en marche d'une broche, prise d'image, etc...). Une action peut être utilisée dans plusieurs opérations : les déplacements d'axe peuvent servir lors d'une opération d'usinage, mais aussi lors d'une opération de contrôle par vision si la caméra est embarquée sur un axe.

## **2.2. Le niveau cellule**

La cellule doit pouvoir gérer l'exécution de plusieurs gammes à la fois. Nous considérons la cellule comme une machine virtuelle à (n) emplacements de travail ou postes de travail, chacun étant capable d'effectuer une sous-phase. Le concept de **poste** permet de **dématérialiser le niveau machine** [MUN 88] (c'est pourquoi la définition de la phase qui représente la machine a été enlevée de l'ensemble des définitions des processus opératoires). Un poste est caractérisé par son état : vide, chargé, au travail, hors service... Le pilotage de niveau cellule gère les flux de produits entre postes en ligne (cf. figure 3.1.), il détermine :

- l'ordre dans lequel on exécute les sous-phases (quand il y a plusieurs gammes à exécuter),
- le lieu où est effectuée chaque sous-phase,

en fonction :

- de règles de gestion déterminant le fonctionnement externe de la cellule,
- des gammes d'usinage spécifiant les sous-phases à effectuer sur les pièces,
- de l'état de la cellule, c'est à dire de l'ensemble des postes.

Le poste est une structure informationnelle permettant de coordonner le niveau cellule et le niveau équipement. Ces fonctions sont :

- synchroniser l'échange de pièces avec d'autres postes,
- exploiter les sous-phases et émettre des demandes de transformations du produit au niveau station,
- gérer son état, et l'état des pièces qu'il supporte.

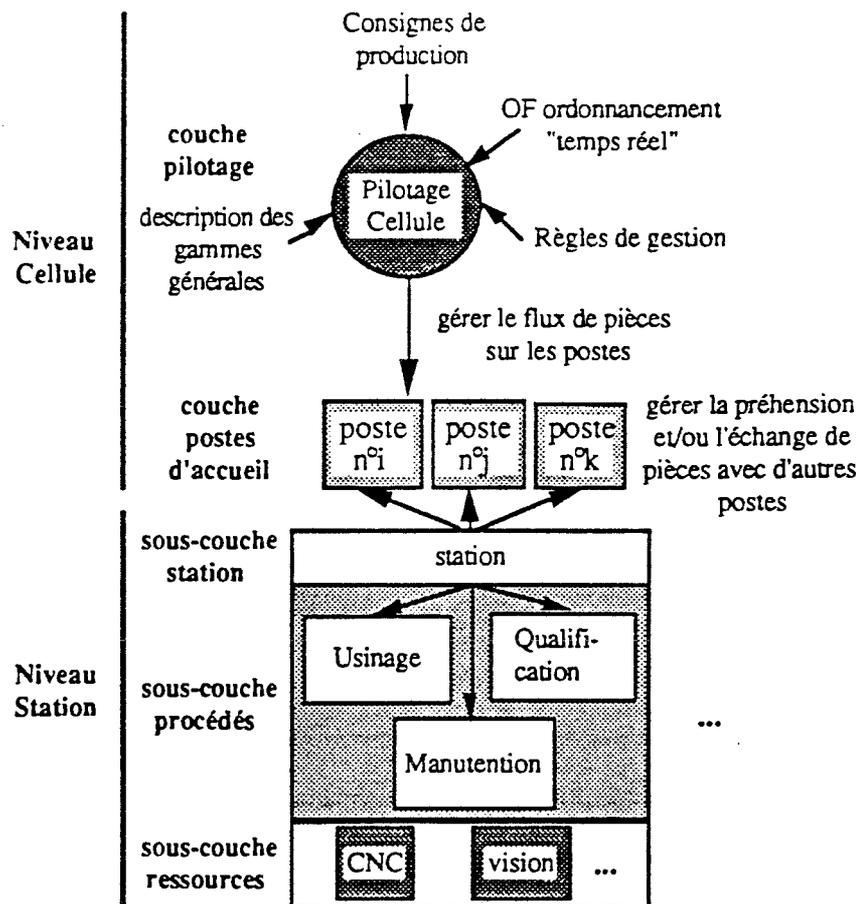


figure 3.1. Architecture fonctionnelle des pilotages cellule et station [RIC 91]

### 2.3. Le niveau station

Le niveau station permet de synchroniser un ensemble d'équipements pour la réalisation d'une ou plusieurs sous-phases. Ces équipements sont :

- une machine de base : tour, centre d'usinage, robot...,
- péri-équipements ou ressources : porte-pièce, outils, capteurs; les péri-équipements peuvent être propres à la station ou partagés avec d'autres stations; un système de vision "multi-utilisateurs" est considéré comme un péri-équipement partagé entre les stations.

La station possède la maîtrise des trois procédés : Transformation, Qualité, Manutention (TQM). Elle possède en outre l'autonomie et l'intelligence quant à la prise de décision concernant l'exécution des sous-phases. Nous proposerons au chapitre IV une méthode et des outils de prototypage des processus opératoires de niveau station (sous-phase) intégrant le procédé de vision.

### 3. Le rôle d'un système de vision dans la cellule

*"Dans ses activités de production, l'homme consacre une part importante de ces efforts à la manipulation et au contrôle d'objets, pièces et outils. L'automatisation de toutes ces activités de production tend à éloigner l'homme des machines" [VOG 90]. Il est donc normal de compenser ce manque de "contrôle humain" par un "accroissement du contrôle automatique" ou par un "accroissement de l'autonomie des machines".*

Cet accroissement d'autonomie passe par :

- le développement des **capacités sensorielles** des machines,
- le développement des **capacités de raisonnement** des machines.

*"Toute tentative de création de machines autonomes pouvant accomplir des tâches multiples dans un environnement mouvant exige que l'on donne à ces machines la possibilité de voir les objets du monde extérieur mais aussi de reconnaître, de comprendre ce qu'elles verront" [TIM 88].*

La vision permet d'augmenter les capacités sensorielles des équipements, et donc d'augmenter leur autonomie.

### **3.1. La vision comme poste de contrôle de la pièce dans la cellule**

Le système de vision peut être envisagé comme un poste à part entière dédié au **contrôle de la pièce**, au même titre qu'un poste de machine à mesurer. Elle permet alors de participer à l'auto-contrôle de la cellule. L'introduction d'un tel poste de contrôle remet en cause le dimensionnement de la cellule et modifie considérablement les flux de produit.

Tel peut être le cas d'un poste de brut installé en début de flux et qui serait une machine destinée à accueillir des produits et fournir des informations de type "identification et localisation" en vue d'une saisie par un robot.

### **3.2. La vision comme outil de contrôle de la pièce et des processus opératoires sur une machine**

Le système de vision est "installé" sur une machine "hôte" non dédiée à la vision (un centre d'usinage, un tour); elle est alors considérée comme un outil de contrôle disponible pour la machine, au même titre qu'un palpeur *renishaw*.

Le système de vision est utilisé comme un moyen de **contrôle des processus opératoires** soit (auto-contrôle des équipements) :

- à travers un contrôle direct de l'outil de fabrication : contrôle d'usure de plaquette [VOG 90]), contrôle de la présence de la pièce dans le montage d'usinage, contrôle de serrage, etc...,
- à travers un contrôle indirect effectué sur le "résultat" du processus opératoire (le produit usiné : la pièce) : vérification de la présence d'un usinage.

Le système de vision est utilisé comme un moyen de **contrôle du produit** par un contrôle direct de celui ci dans la machine.

### 3.3. La vision "réflexe"

Utilisés pour l'asservissement en temps réel d'un axe ou d'un système poly-articulé (robot), ces systèmes sont utilisés dans des applications à contrainte temps-réel nécessitant des architectures spécifiques et ne permettant pas un découplage de la commande du système de vision esclave de celle du système utilisateur. Quelques exemples en sont : le soudage électrique par guidage laser, l'asservissement en position de lampes nécessitant des temps de traitement très rapides [LEV 88], le contrôle temps réel de plaques de PMMA sur tapis roulant [WOL 90].

## 4. Objectifs pour la vision

Nous faisons dans ce paragraphe une liste informelle de ce que nous pensons que doivent être les qualités actuelles d'un système vision intégré dans un environnement de cellule flexible.

### 4.1. Un capteur flexible, multi-fonctions, mutli-utilisateurs

Le concept de flexibilité se définit par [Mun 88] :

- *la flexibilité de fabrication* : la capacité à produire simultanément différents types de produits en s'adaptant aux variations des entrées, aux ordres d'un niveau supérieur (production, planification), aux modifications de l'environnement physique (aléas, pannes) et aux changements de conditions (modes de marche, retrait d'une machine pour maintenance),
- *la flexibilité de reconfiguration* : la capacité à changer rapidement de production, aussi bien au niveau machine qu'au niveau atelier,
- *la flexibilité de génération* : la capacité à modifier une installation (adjonction, retrait ou échange d'équipements et fonctions) sans impliquer une révision complète de l'ensemble du système.

Le coût relativement élevé d'une machine vision peut être compensé par une utilisation partagée entre plusieurs postes de travail (le temps d'utilisation moyen par poste est faible par rapport aux temps d'usinage et de manutention); cet

aspect multi-fonctions impose les qualités de flexibilité définies plus haut car le même système de vision peut traiter plusieurs problèmes dans une même cellule.

Le système de vision doit être "l'oeil de la cellule", capable d'aller chercher des informations non dédiées desquelles il est possible d'extraire des données relatives aux variations d'environnement, participant ainsi à l'autonomie de la cellule.

## **4.2. Un capteur intégré**

### **4.2.1. Aspect communication**

La contrainte de partage de la ressource vision nous amène à différencier les fonctions de vision pure des fonctions d'interface avec les équipements d'usinage ou de manutention coopérants. D'un point de vue organique (organisation du système physique), ces fonctions d'interface avec l'environnement (contrôle-commande, partage et transfert de données) doivent être supportées par des tâches coopérantes dans un système temps réel. Ces tâches supplémentaires peuvent être réalisées par l'emploi de réseaux.

### **4.2.2. Aspect contrôle commande**

L'architecture interne de contrôle-commande d'un système de vision intégré doit être compatible avec l'architecture fonctionnelle de contrôle-commande de son environnement. L'architecture hiérarchique distribuée de la cellule telle que nous l'avons définie impose aux éléments du système de vision d'être autonomes et intelligents et de permettre la commande d'opération de vision depuis le niveau station.

## **4.3. Un capteur auto-contrôlé**

L'amélioration de la qualité en fabrication passe par l'auto-contrôle de la production et l'auto-contrôle des moyens de production afin d'améliorer leur **sûreté** et leur **autonomie** de fonctionnement, leur sécurité et leur maintenabilité par l'utilisation accrue de capteurs (utilisation d'un capteur de vibrations ou de couple pour détecter un dysfonctionnement lors d'une opération d'usinage) et par

une meilleure structuration des parties commandes associées aux différentes parties opératives [SFA 89].

Cette notion d'auto-contrôle doit être appliquée à tous les niveaux d'équipements de la cellule en partant aussi bien d'un composant aussi trivial que *"l'actionneur tout ou rien qui peut être source tant de non-productivité que de non-qualité si l'on ne peut garantir sa fiabilité"* [MOR90] jusqu'au processus d'usinage.

Les principales fonctions d'auto-contrôle sont :

- analyse de la cohérence d'une demande par rapport à l'état du capteur,
- gestion de la disponibilité de moyens,
- surveillance du déroulement de l'opération,
- analyse de la pertinence des résultats,
- intégration de procédures d'auto-diagnostic.

#### **4.4. Un capteur ouvert**

Un capteur "intégrable" doit pouvoir être ouvert à son environnement. Les principaux échanges que réalise un système automatisé avec son environnement sont :

- échange de données,
- dialogues de type "contrôle/commande" avec d'autres équipements,
- dialogues de type "exploitation" avec l'opérateur, il s'agit de l'aspect maintenance et gestion technique.

**L'intégration des données du système de vision dans le Système d'Informations Industrielles de l'entreprise est un facteur majeur d'intégration.** Cette intégration par les données est à la base de l'intégration dans le système de gestion. Les activités de conception sont consommatrices de nombreuses données : matériel de vision disponible, définition du produit, savoir faire en visionique, etc ...

Le système physique de l'entreprise est lui aussi tributaire de l'organisation des données dans chaque domaine et de leurs aptitudes à être réparties (gammes de contrôle, résultat d'un contrôle, fichier image, etc ...).

Il est important de spécifier l'ensemble des données "utilisées, fabriquées, consommées" par la vision.

#### 4.5. En bref : les yeux de la cellule

L'implantation de toutes ces qualités : flexibilité, auto-contrôle, maintenance, ouverture, doit conduire à la réalisation d'un système de vision plus "facile" à mettre en oeuvre (smart sensor).

*"The development of new measurement capabilities such as easy adjustment for operating range, compensation for effects of environmental conditions and self-diagnosis, has given a considerable level of "intelligence" to current sensors which are smarter, smaller and cheaper than ever used before" [HEI 91].*

Notre objectif est de faciliter l'utilisation (l'intégration) des capteurs sans contact de façon à augmenter leur nombre et la **redondance d'informations** au niveau de la cellule. Le système de vision peut devenir "l'oeil de la cellule", multi-fonctions, multi-utilisateurs. **Cette "facilité" peut modifier les habitudes de l'utilisateur** qui hésitait jusqu'à présent à installer un système de vision et peut augmenter le rôle de celle ci dans la cellule.

**La complexité des traitements de vision ne doit pas être un obstacle à la "prise d'information" sur le site** (prise d'image); cette information, même si elle n'est pas traitable par un système informatique, peut simplement être stockée pour être utilisée ultérieurement par un opérateur humain.

L'utilisation répartie du système de vision peut avoir des répercussions sur la conception des systèmes de vision qui ont trop tendance à réunir sur un même équipement et en une même opération à la fois acquisition d'information et traitement de celle ci. Les systèmes de vision doivent être modulaires. **La séparation de l'acquisition du signal et de son traitement** est primordiale pour la réalisation d'un capteur intégré.

Cette séparation impose des moyens de transfert et de stockage du signal (image) à travers la cellule.

## **5. Décomposition fonctionnelle**

Nous proposons un *Modèle de Spécifications Fonctionnelles de Référence* pour "réaliser une opération de vision" dans un environnement de cellule flexible (figure 3.2.). Son objectif est de montrer les fonctions d'un système de vision. Ce modèle doit servir de référence à la description de cas particuliers; nous l'utiliserons pour faire la décomposition fonctionnelle d'un "contrôle de perçage sur fraiseuse CHARLY-ROBOT".

### **5.1. Fonction préparer triplet "source-objet-recepteur"**

Nous avons montré précédemment que les éléments du système d'acquisition d'informations d'un système de vision sont la **source, l'objet et le récepteur**; pour qu'une opération de vision se déroule correctement, il faut "préparer" ce triplet. Par exemple : allumer une lampe, nettoyer l'objet, le tourner d'une certaine position, ouvrir un sas, amener la caméra, régler les objectifs, etc ... Cette fonction de préparation ne fait pas seulement intervenir du matériel de vision (lampe, caméra, etc...) mais aussi des équipements de l'environnement tels que les axes d'une machine outil si la caméra est montée dessus, ou un mécanisme de positionnement spécifique à l'objet et à la position que l'on désire obtenir pour l'opération de vision.

### **5.2. Fonction capter information**

Cette fonction constitue le cœur de l'acquisition d'informations; elle peut être comparée à l'exécution d'un programme d'usinage sur une machine outil à commande numérique. En effet le rôle d'un programme d'usinage est de commander un déplacement relatif entre l'outil et la pièce en respectant des consignes précises. De même, la prise d'information en vision nécessite le positionnement relatif de la caméra (l'outil), de l'objet (la pièce) et de la source d'éclairage selon des modalités bien définies.

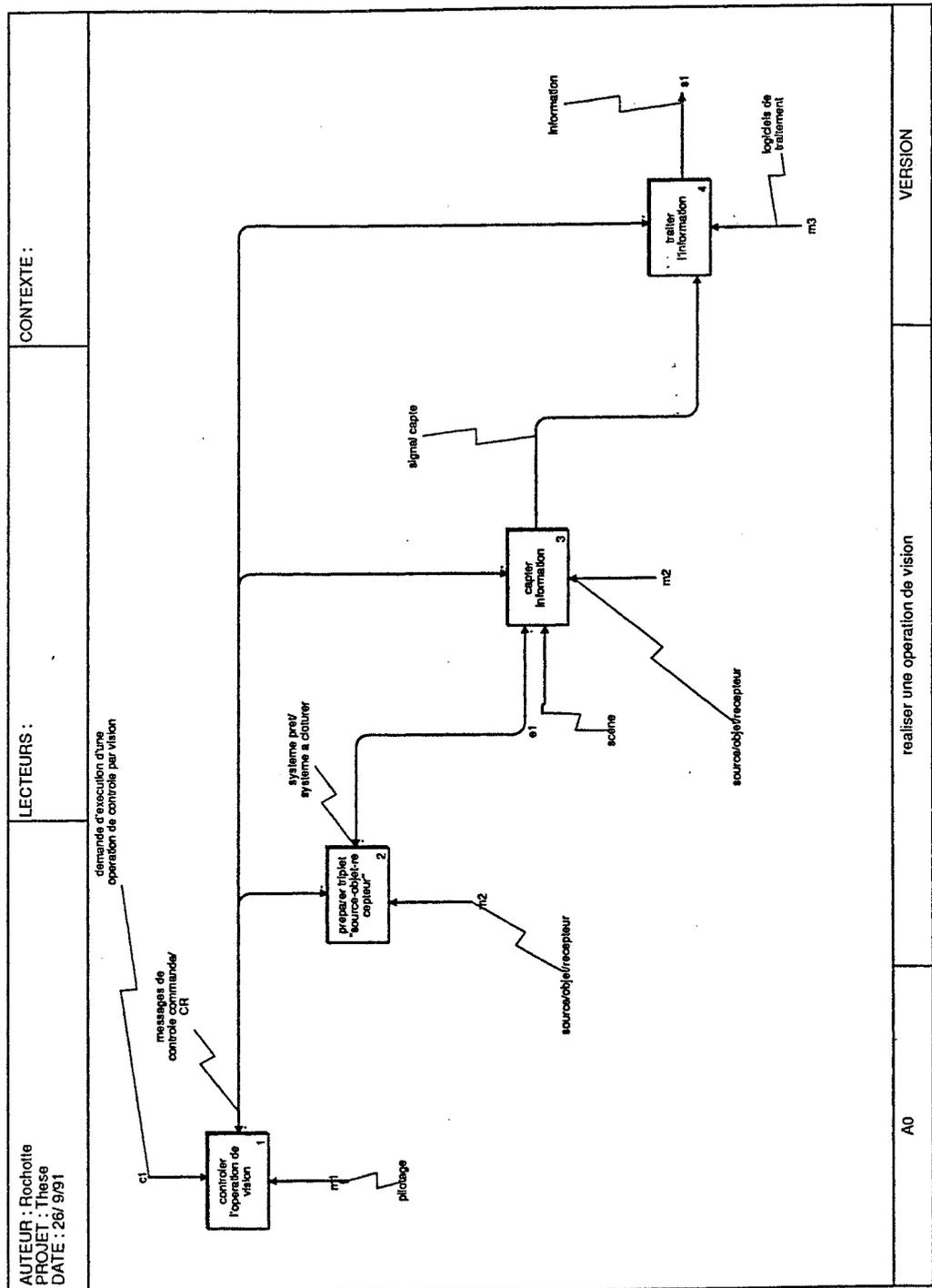


figure 3.2. SADT de Spécifications Fonctionnelles de Référence : réaliser une opération de vision

Dans le cas de l'utilisation d'un système linéaire, la caméra ou l'objet doivent être en mouvement l'un par rapport à l'autre; dans le cas d'un système de vision matricielle, la caméra peut être fixe par rapport à l'objet observé; dans le cas d'un système de vision tridimensionnel, la source laser doit balayer la scène.

La prise d'information n'est donc pas triviale.

### **5.3. Fonction traiter l'information**

Cette fonction constitue l'ensemble des transformations que l'on fait subir à l'information brute issue du capteur (numérisation, prétraitement, segmentation, interprétation).

Cette activité est purement informatique et peut être effectuée sur un matériel totalement indépendant de la prise d'image.

### **5.4. Fonction contrôler l'opération de vision**

Cette fonction consiste à réaliser la synchronisation des activités précédentes : "préparer/cloturer le triplet source/objet/recepteur", "capter l'information", "traiter l'information".

Cette activité est supportée, dans le cadre de référence physique de la cellule, par le niveau station.

### **5.5. Les données nécessaires à l'exécution d'une opération de vision**

L'exécution d'une opération de vision met en oeuvre différents mécanismes (exécutables) de commande. Ces mécanismes sont tous des *sous-types* de *l'individu* PROGRAMME TACHE.

Une opération de vision est en relation avec un certain nombre de programmes tâche qui peuvent intervenir à la fois sur l'objet, la source ou le récepteur, en préparation, en action ou en clôture (cf. figure 3.3.).

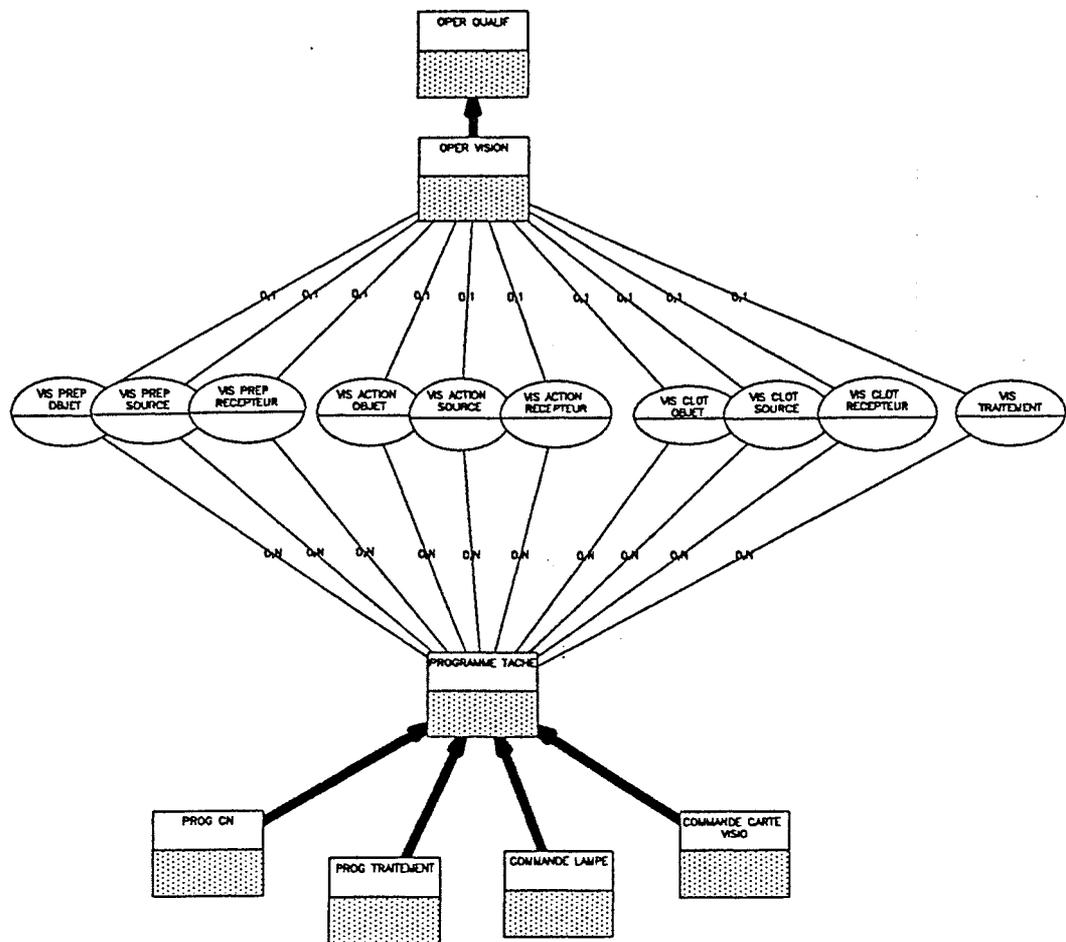


figure 3.3. Extrait du MCD de référence de vision : opération de vision

## 6. Les objets "intelligents" en automatique

Nous voulons conférer aux modules qui composent un système de vision une structure telles qu'ils soient autonomes et intelligents. Il nous est apparu nécessaire de faire le point dans ce domaine.

La notion "d'objet intelligent" est née de la rencontre de deux tendances :

- le besoin de décentraliser la commande des systèmes, l'ensemble des fonctions d'une application n'étant plus supportable par un seul équipement; cette tendance est à la base de la répartition des traitements et de la décentralisation des matériels,

- le besoin méthodologique, qui par la volonté de réutilisation, débouchait naturellement sur la notion d'objet (modèle générique de commande) capable "d'encapsuler la connaissance", servant de brique à la conception de la partie commande d'une application.

Cette volonté de décentralisation a amené les constructeurs à fabriquer des réseaux capables de répartir les traitements afin de favoriser la communication entre les différents modules. La "multiplication" des processeurs de traitement (grâce aux réseaux) a permis une conception de la partie commande intégrant plus de traitements, donc plus d'intelligence. D'autre part, la notion de module physique décentralisé se marie parfaitement avec la notion d'objet informatique réutilisable en Génie Logiciel.

### 6.1. Le concept de filtre

La notion de filtre de comportement est apparue avec la volonté d'appliquer la modélisation de la partie opérative à la structuration de la partie commande [LHO 85] [ALA 86]. Les composants qui en résultent doivent pouvoir être réalisés pour la description de la partie commande (PC) par la définition d'Eléments de Partie Opérative (EPO) fournissant, chacun, une image d'un comportement élémentaire.

La modélisation de la partie opérative peut être utilisée pour :

- *émuler* le comportement de la partie opérative dans les phases de test; cette approche est développée dans l'outil ADELAIDE [COR 89] (émulateur de partie opérative); elle permet d'obtenir une meilleure connaissance des éléments opératifs et d'autre part améliore la qualité du travail de développement,
- *filtrer* les échanges entre la Partie Commande et la Partie Opérative; ce filtrage permet de détecter que les ordres reçus de la PC sont compatibles avec l'état de la PO et que les retours issus de la PO (compte rendu des capteurs) sont compatibles avec ceux prévus par le modèle; le filtre permet donc la "surveillance fonctionnelle de la PC" et la "surveillance opérationnelle de la PO". Ce système "soulage" les modules supérieurs de commande de la

surveillance, il évite la remise en cause des tests à chaque modification des niveaux de commande supérieurs et il évite la propagation des erreurs.

Le filtre stocke la connaissance opérationnelle d'un niveau fonctionnel donné et gère les activités de ce niveau (commande de PO et surveillance). Ce "savoir-faire" est mis à la disposition des niveaux fonctionnels d'ordre supérieur qui n'ont pas à s'investir dans les niveaux inférieurs. Ce principe participe à la décentralisation des traitements et inspire une **généralisation** du concept d'Elément de Partie Opérative à tous les niveaux fonctionnels d'organisation. Cette généralisation implique une approche modulaire et hiérarchisée de la conception de la structure fonctionnelle de l'automatisme.

## 6.2. Le module fonctionnel d'automatisme

Le Module Fonctionnel d'Automatisme (MFA) est né dans le cadre d'une collaboration EDF - Université de Lille; *"le but est d'aboutir à des entités autonomes capables d'assurer des traitements d'une manière intelligible, et ce, à n'importe quel niveau fonctionnel de la structure hiérarchique d'un système"* [BEL 89].

Chaque MFA permet d'assurer de manière autonome le traitement de diverses fonctions issues de l'analyse fonctionnelle du processus.

Le MFA a trois objectifs :

- *contrôle* : analyse et identification des objectifs envoyés par leurs supérieurs d'une part, et des états de leurs subordonnés,
- *commande* : l'élaboration d'une variable de commande : action, ordre ... représentant un objectif pour les niveaux inférieurs,
- *analyse de comportement* : (filtre de comportement) analyse du comportement de leurs subordonnés en fonction des ordres qu'ils ont reçus et des informations d'état qu'ils ont restituées, avec élaboration d'un compte rendu comme constat de cette analyse.

les MFA se composent de quatre parties :

- M.I.O : module d'identification d'objectif : il vérifie la cohérence et la crédibilité des objectifs qu'il reçoit du niveau supérieur,
- M.I.E : module d'identification d'état : il élabore des informations sûres et synthétiques de l'état de tous les subordonnés,
- M.D.C : Module de décision et de commande : génère un ordre validé vers les niveaux inférieurs,
- M.A.C : module d'analyse du comportement : élabore, en fonction des objectifs, des états et des commandes des informations fiables sur le comportement fonctionnel du MFA pour les niveaux supérieurs.

### 6.3. L'actionneur "intelligent"

Un actionneur intelligent doit pouvoir remplir les deux fonctions suivantes [STA 90] :

- agir sur le processus,
- générer et valider les informations relatives à son état.

Les nouvelles fonctionnalités définies par la CIAME et le projet DIAS [IUN 91] sont :

- *mise en service* : la configuration et le paramétrage de l'actionneur intelligent (qui doivent permettre une personnalisation de celui ci), la possibilité de mise en service progressive en l'absence d'environnement de contrôle-commande;
- *conduite* : la possibilité pour les opérateurs de conduire l'actionneur en mode local, de consulter ou de modifier des paramètres; la possibilité de gérer des modes de marches; accroître la qualité des informations sur l'état réel de l'actionneur et ses disponibilités;
- *maintenance* : curative : par la détection et la localisation des défauts et l'aide au diagnostic; préventive systématique : dans la comptabilisation de sollicitations; préventive conditionnelle : par la surveillance des dérives;
- *gestion technique* : les actionneurs intelligents permettent d'effectuer des statistiques de panne, de dégradation, de consommation d'énergie.

#### **6.4. Le capteur "intelligent"**

Le capteur "intelligent" est une partie de l'actionneur "intelligent". Il peut utiliser un certain nombre de parties opératives pour réaliser sa mesure. C'est notamment le cas d'un système de vision où la mesure nécessite l'allumage de lampes, le déplacement d'une caméra, le réglage d'un zoom motorisé, etc ... Les fonctionnalités définies pour l'actionneur "intelligent" sont donc effectives pour le capteur "intelligent".

L'objectif propre du capteur "intelligent" est d'accroître la **crédibilité** et l'**exactitude** de sa mesure. Pour ce faire, on définit trois niveaux de validation :

- *validation technologique* : il s'agit de vérifier le bon fonctionnement de l'ensemble des composants du capteur lors de la mesure. Il faut pour cela disposer de capteurs sur le capteur,
- *validation opérationnelle* : il s'agit de comparer la mesure effectuée avec une ou plusieurs mesures redondantes réalisées sur d'autres capteurs,
- *validation fonctionnelle* : on vérifie que la mesure effectuée est cohérente avec l'environnement fonctionnel du process mesuré. Pour ce on peut utiliser d'autres mesures et une loi qui modélise le bon fonctionnement du process.

### **7. Application à l'intégration physique d'un système de vision dans une cellule flexible d'usinage**

#### **7.1. L'architecture physique de la cellule**

L'architecture physique de la cellule que nous avons réalisée dans le cadre du projet MRT [MRT 91] repose sur l'utilisation de trois réseaux (cf. figure 3.4.) :

- le réseau ETHERNET couplé avec SQL pour la gestion et le transfert des informations dans la cellule,
- le réseau MAP pour la répartition des ordres de contrôle-commande de niveau cellule (échanges entre station),

- le réseau BITBUS pour la communication avec les équipements de niveau machine.

L'ensemble des équipements qui ont accès à la base de données sont connectés au réseau ETHERNET. Ainsi l'ordinateur cellule, le PS de la station CHARLY-ROBOT, le PS de la station POSTE DE BRUT et l'AT de vision sont connectés à la base de données qui est physiquement implantée sur un serveur (PS).

#### **7.1.1. L'ordinateur cellule**

L'ordinateur cellule supporte le noyau de pilotage de la cellule. Il communique avec la station CHARLY-ROBOT et la station POSTE DE BRUT via le réseau MAP. Le logiciel de pilotage gère les flux entre postes grâce à une vue dématérialisée des postes supportée par les VMD (Virtual Manufacturing Device) prévus dans la couche application de MMS.

#### **7.1.2. La station du CHARLY-ROBOT**

Le poste d'usinage CHARLY-ROBOT doit effectuer les différentes opérations d'usinage sur la pièce NASA. Le PS de la station CHARLY-ROBOT est connecté par une liaison RS232 à ses équipements propres qui sont le LPC, l'automate et la NUM760.

La table CHARLY-ROBOT est une fraiseuse didactique à trois axes (XYZ) sur laquelle est monté (sur l'axe Z) une broche, un palpeur RENISHAW et une caméra CCD miniature SYSTEME SUD (de cette façon elle peut être déplacée en tout point de la table de travail, on peut donc observer à volonté à la fois le montage d'usinage et le produit qu'il contient); un montage d'usinage et trois lampes halogènes font également partie des ressources propres de la station CHARLY-ROBOT. La station est connectée au réseau BITBUS de façon à pouvoir communiquer avec la commande du porte-pièce, la commande du système d'éclairage et la commande de la caméra.

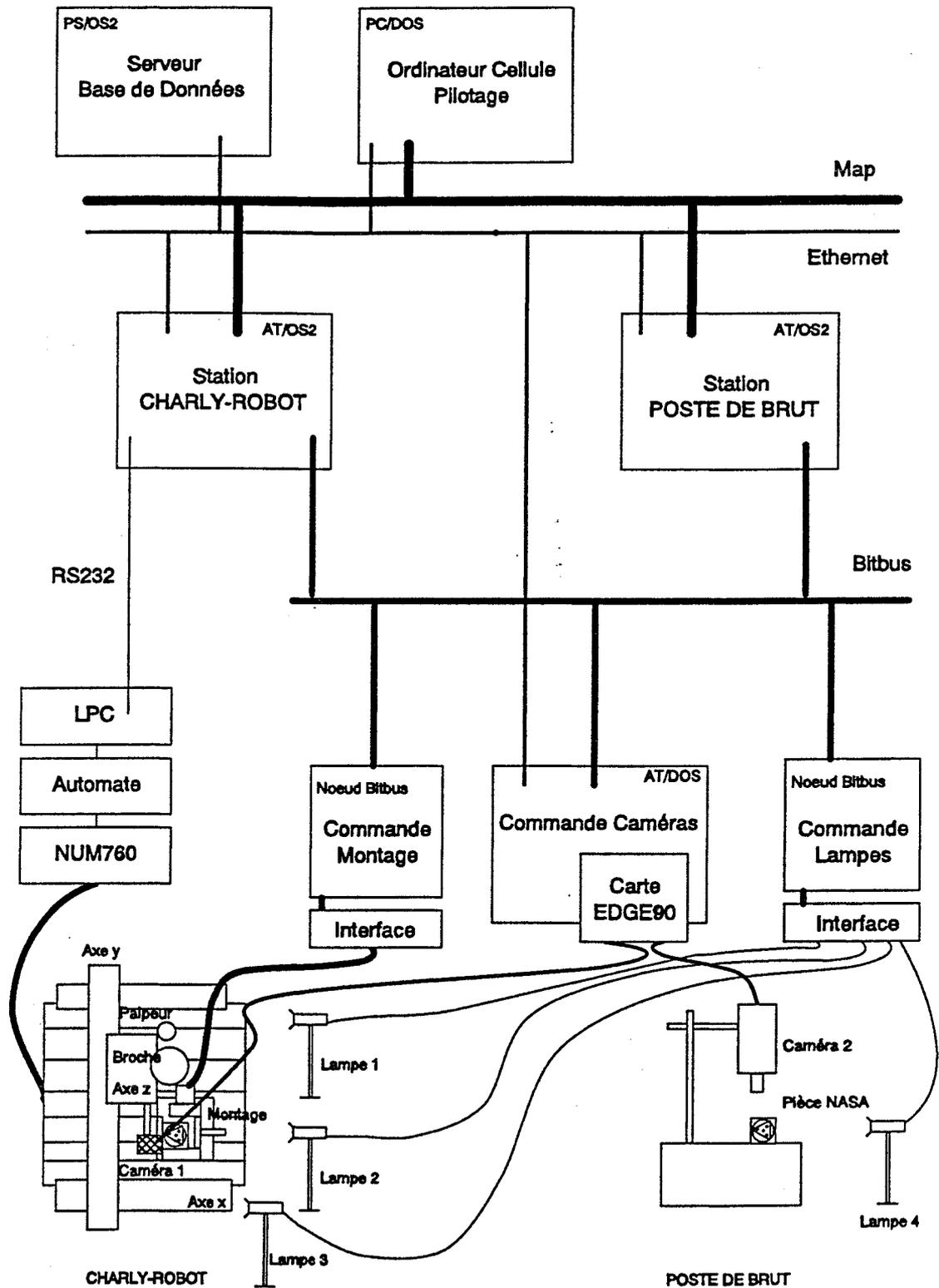


figure 3.4. Architecture physique de la cellule de la maquette MRT

### **7.1.3. La station POSTE DE BRUT**

La station POSTE DE BRUT a plusieurs objectifs : détecter la présence d'un objet sur la surface du poste, vérifier qu'il s'agit bien d'un brut de la pièce NASA et calculer sa position en vue d'une saisie par un robot (bien que notre maquette expérimentale ne dispose pas encore de robot manipulateur).

Pour effectuer tous ces contrôles, le poste de brut utilise exclusivement le système de vision. Il dispose pour cela d'une caméra video et d'une lampe halogène dont les commandes sont accessibles sur le réseau BITBUS.

### **7.1.4. Les équipements partagés**

Les équipements partagés sont tous connectés au réseau de terrain BITBUS. Seul le système de vision avec ses éclairages est partagé : la commande des quatre lampes halogènes de la cellule se réalise dans un seul noeud BITBUS et la commande des deux caméras de la cellule est implantée sur un AT (carte EDGE90) connecté au réseau. Ainsi seules les commandes des lampes et la carte de vision sont partagées.

La commande du montage d'usinage a été implantée sur un noeud BITBUS par pure commodité, cet équipement n'est pas fonctionnellement partageable.

## **7.2. Les contrôles réalisés**

### **7.2.1. Les contrôles sur le POSTE DE BRUT**

Les opérations de vision sur le POSTE DE BRUT sont :

- test de présence,
- identification et localisation de l'objet NASA.

La station "commande" cycliquement une prise d'image sur le poste de brut et réalise le traitement "test de présence"; si une pièce est présente, la même image est utilisée pour le traitement "identification/localisation".

### **7.2.2. Les contrôles sur le poste d'usinage CHARLY-ROBOT**

Le système de vision nous permet d'augmenter l'auto-contrôle du niveau station en assurant une redondance d'informations.

Les opérations de vision sur la station CHARLY-ROBOT sont :

- test de serrage sur le montage d'usinage,
- test de présence de l'objet NASA dans le montage d'usinage,
- contrôle de présence et de dimension d'une entité de perçage sur la pièce NASA.

Chaque opération de vision possède sa propre configuration physique aussi bien pour l'éclairage, la position de la caméra que les traitements appliqués. Nous avons choisi de détailler l'opération : "contrôle d'un perçage".

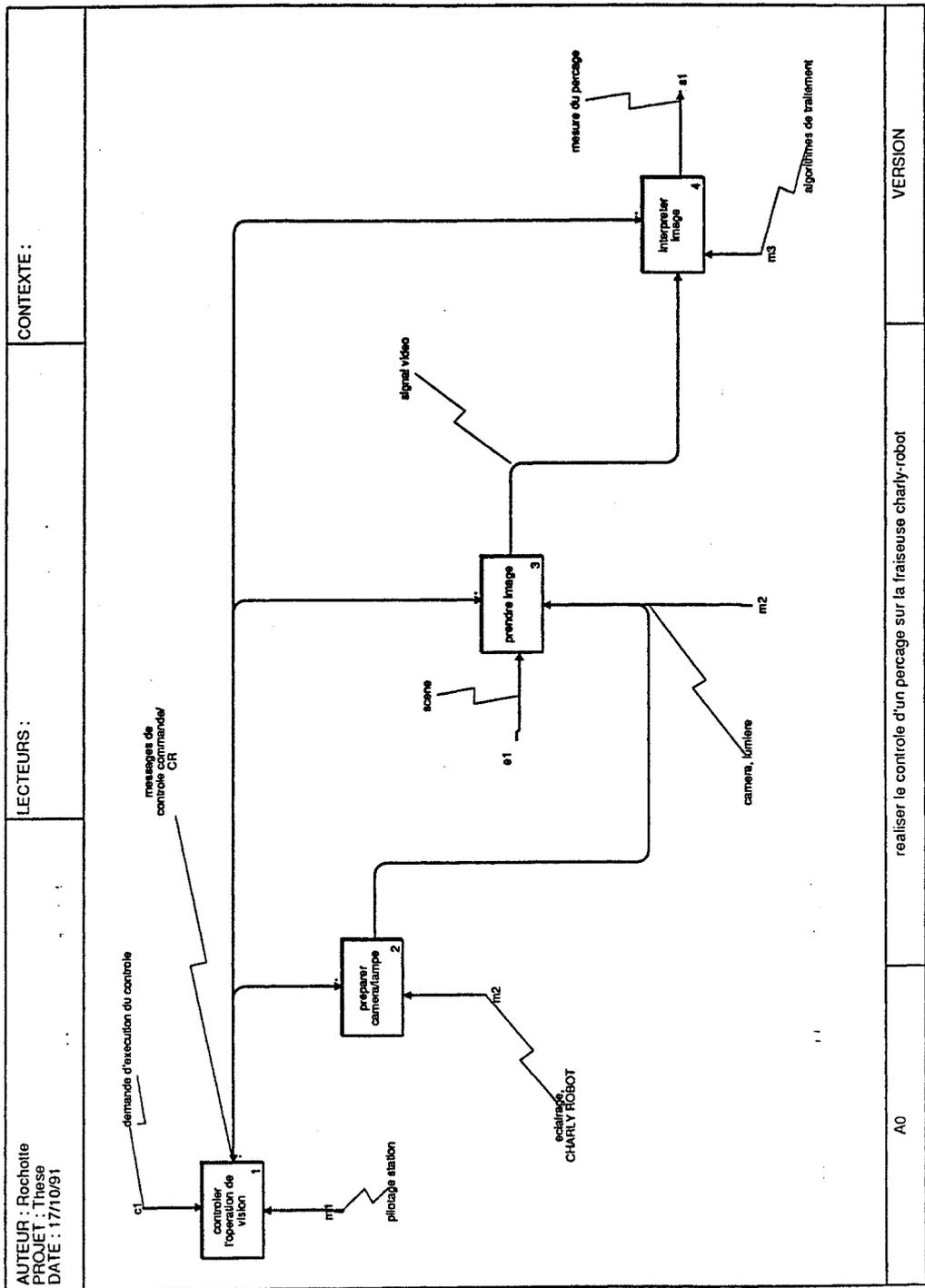
### **7.3. Exemple : contrôle d'un perçage sur fraiseuse CHARLY-ROBOT**

Nous utilisons le Modèle de Spécifications Fonctionnelles de Référence "réaliser une opération de vision" (cf. figure 3.2.) pour concevoir un Modèle de Spécifications Fonctionnelles Particulier "réaliser le contrôle d'un perçage sur la fraiseuse charly-robot" (cf. figure 3.5.).

A partir de ce dernier, nous réalisons un Modèle Particulier (SADT) de Spécification/Conception (cf. figure 3.6.) (cette étape permet de définir les différents modules matériels).

Nous utilisons ensuite l'outil ORCHIS-GRILLE pour définir les interfaces d'entrée-sorties de diagrammes fonctionnels dont le comportement interne doit être validé ensuite par l'outil SPEX (cf. figure 3.7.).

Notre démarche est structurée, elle fait appel à différents outils de l'ingénierie productique tels que ORCHIS, ORCHIS-GRILLE et SPEX.



*figure 3.5. SADT de Spécifications Fonctionnelles Particulier : réaliser un contrôle de perçage sur une fraiseuse CHARLY-ROBOT*

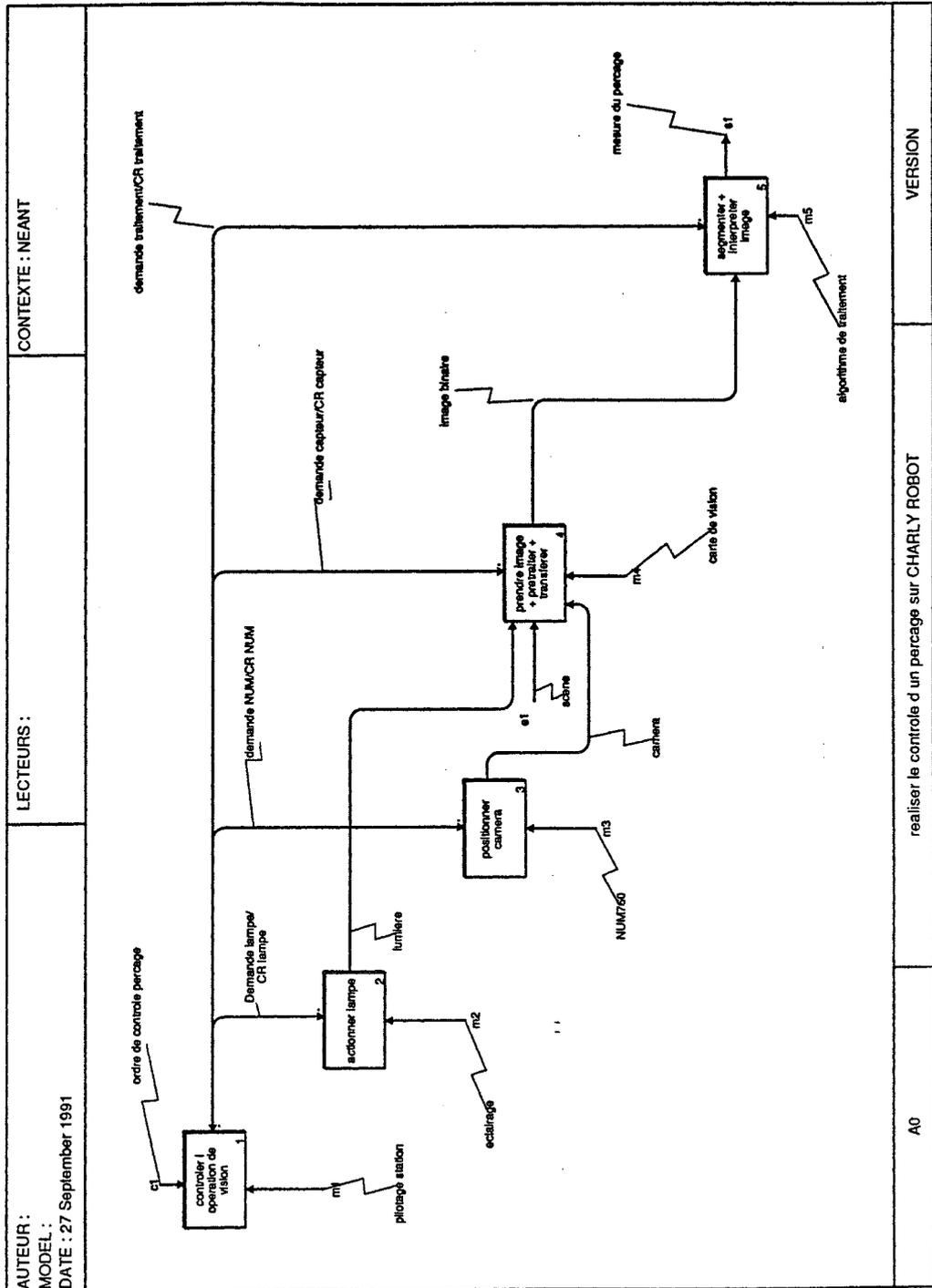


figure 3.6. SADT de Spécifications/Conception : réaliser un contrôle de perçage sur une fraiseuse CHARLY-ROBOT

AUTEUR :		LECTEURS :				CONTEXTE : contrôler / opération de vision					
MODEL :		A1		A2		A3		A4		A5	
DATE : 27 September 1991		A1		A2		A3		A4		A5	
titulaire	concret	A1		A2		A3		A4		A5	
concret	réaliser le contrôle d'un perçage sur CHARLY ROBOT	A1		A2		A3		A4		A5	
A1	contrôler / opération de vision	A1		A2		A3		A4		A5	
A2	actionner lampe	A1		A2		A3		A4		A5	
A3	positionner camera	A1		A2		A3		A4		A5	
A4	prendre image + prétraiter + transférer	A1		A2		A3		A4		A5	
A5	segmenter + interpréter image	A1		A2		A3		A4		A5	
A0		réaliser le contrôle d'un perçage sur CHARLY ROBOT				VERSION					

figure 3.7. GRILLE : réaliser un contrôle de perçage sur une fraiseuse CHARLY-ROBOT

### **7.3.1. Fonction "actionner lampe"**

Cette fonction regroupe un ensemble d'activités : allumer /éteindre la lampe, surveiller l'état des lampes etc... Ce module doit avoir une structure d'objet intelligent.

#### **7.3.1.1. Prototypage**

Nous avons prototypé la commande du module d'éclairage sur SPEX en utilisant le fichier d'entrée/sortie généré avec ORCHIS-GRILLE. Son comportement a été décomposé selon le Modèle de Référence proposé dans [LHO 85] : PILOTER, CONDUIRE, COORDONNER (surveiller, réagir, commander) (cf. figure 3.8.).

#### **7.3.1.2. Réalisation**

Le module d'éclairage a été implanté sur un noeud BITBUS; il gère la commande de quatre lampes simultanément. Pour effectuer la surveillance et la détection de défauts sur les lampes, des capteurs photo-sensibles ont été fixés sur les projecteurs afin de disposer d'une information en retour pour le noyau de surveillance.

Le filtre a été programmé en PLM51 sur EPROM (cf. figure 3.9.). Nous avons seulement réalisé le module coordonner, il permet :

- d'allumer, d'éteindre une lampe au choix,
- de filtrer les commandes absurdes (allumer la lampe déjà allumée...)
- de renvoyer des défauts : de commande, d'allumage, d'extinction,
- de renvoyer l'état des lampes,
- d'éteindre systématiquement une lampe allumée plus de quatre secondes (les ampoules halogènes ont une faible durée de vie).

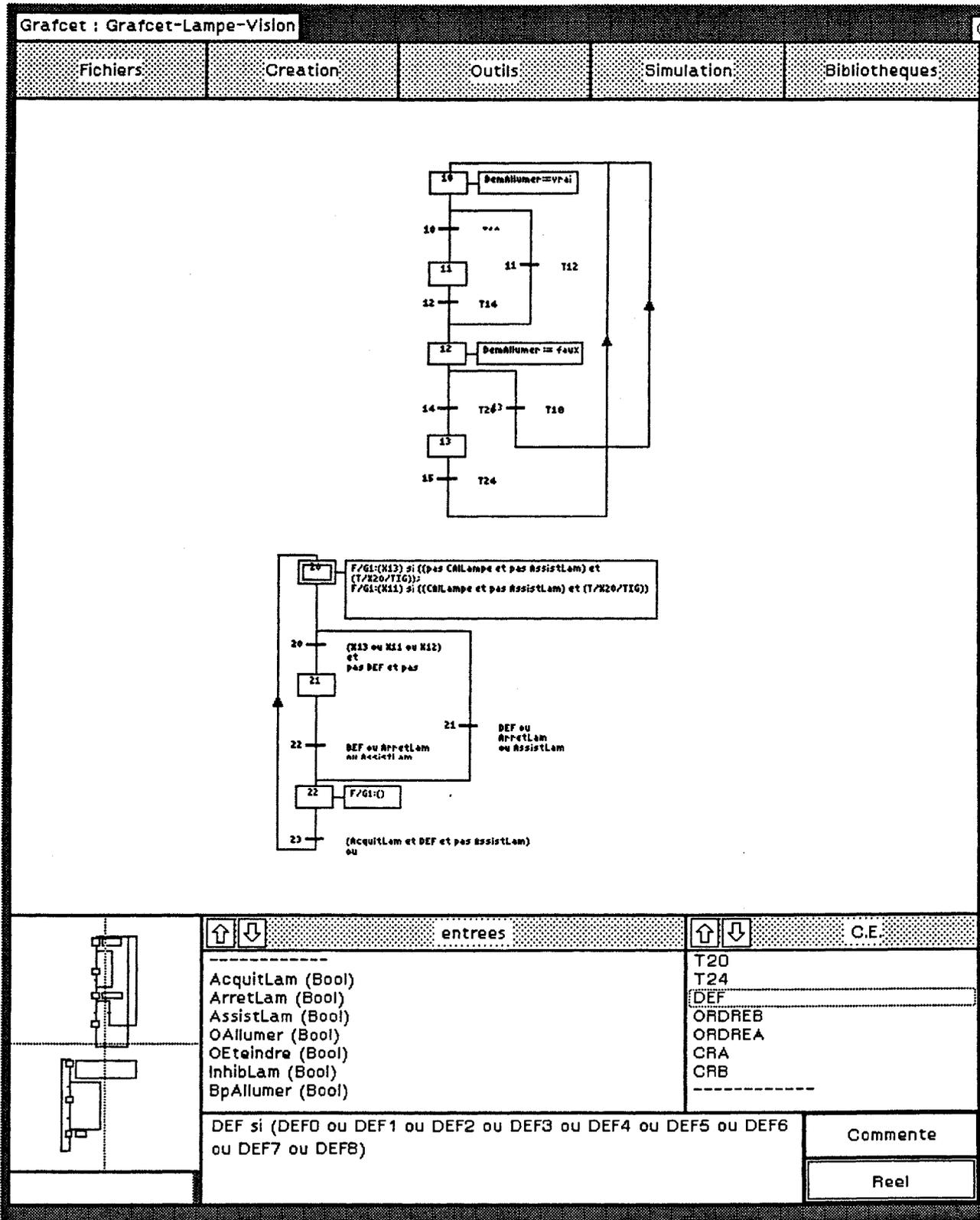


figure 3.8. Modèle de Filtre prototypé sur SPEX : modules commander, surveiller la lampe

```

/* description des erreurs de fonctionnement */
#erreur1:
if (etape1=1) and ((capteur and 1h)=0) then
do;
com1(2)=0e2h; /* default extinction */
com1(3)=(com1(3) and 3fh) or 40h; /* default de fonctionnement */
end;
if (etape1=1) and ((capteur and 1h)=1) then
do;
com1(2)=20h; /* eteint pas de default */
end;

if (etape1=3) and ((capteur and 1h)=1) then
do;
com1(2)=0e1h; /* default allumage */
com1(3)=(com1(3) and 3fh) or 40h; /* default de fonctionnement */
end;
if (etape1=3) and ((capteur and 1h)=0) then
do;
com1(2)=0h; /* allumé pas de default */
end;

/*etat capteur lampe */

if ((capteur and 1)=0) then com1(3)=(com1(3) and 0c0h) or 1;
else com1(3)=com1(3) and 0c0h;

if etiq1=1 then
do;
com1(0)=(com1(0) and 0fh) or 20h; /* fin de travail */

```

*figure 3.9. Extrait de programme en PLM51 de la commande des lampes sur un noeud BITBUS*

### 7.3.2. Fonction "positionner caméra"

Cette activité est réalisée par la commande numérique, en effet le corps de la caméra est embarqué sur l'axe Z du CHALY-ROBOT; la caméra est un outil supplémentaire de la NUM760, toute action visant à déplacer celle-ci fait donc intervenir les commandes d'axe de la NUM. Nous avons réalisé un (petit) programme sur la commande numérique pour exécuter l'action : "Positionner Caméra", et un autre pour l'action : "Dégager caméra" (cf. figure 3.10.).

```

X501
N10      (nasa/trou1,amenée)
N20      G52 G77 T3 D3 H9100
N30      G52 G0 X303.25 Y290.253
N40      G52 G01 Z80. F1200
N50      M2

X505
N10      (nasa/trou1trou2trou3trou4,retrait camera)
N20      G52 G77 T3 D3
N30      G52 G0 X303.25 Y304.983 Z0
N40      M2

```

*figure 3.10. Extrait de programme en code NUM760 : Positionner Caméra, Dégager Caméra*

### **7.3.3. Fonction "prendre image + prétraiter + transférer"**

#### **7.3.3.1. Prototypage**

Nous avons prototypé la commande du module de prise d'image sur SPEX en utilisant le fichier d'entrée/sortie généré avec ORCHIS-GRILLE. Son comportement a été décomposé selon le Modèle de Référence proposé dans [LHO 85] : PILOTER, CONDUIRE, COORDONNER (surveiller, réagir, commander).

#### **7.3.3.2. Réalisation**

Le module de prise d'image a été implanté en langage C sur un AT/DOS/RTC (RTC : RealTime Craft) (cf. figure 3.11.); il gère la commande de quatre caméras. La carte EDGE90 effectue la prise d'image selon un certain mode de prétraitement (spécifié dans la commande), le transfert de l'image est réalisé par le réseau ETHERNET.

Le filtre permet :

- d'effectuer une commande de prise d'image,

- de filtrer les commandes absurdes (prendre une image avec une caméra qui n'existe pas)
- de renvoyer des défauts : de commande, de caméra, de lecture de fichier de configuration de la carte EDGE 90, d'écriture de l'image sur le réseau, d'initialisation du capteur (lecture du fichier de commandes),
- de renvoyer l'état des caméras.

```

/*****
/* Task 4 : chien de garde caméra */
/*****
void far ChienCam()
{
char *msg;
char *prise;
int num_error;
Word Status;
int default=0;
while (TRUE)
{
/* surveillance de la duree de la prise d'image */
Status = Receive (cChienGardBal,(DoubleWord far *)&prise, cInfiniteWaiting);
Status = Receive (cChienGardBal,(DoubleWord far *)&prise,Duree_Clic);

if (*prise!='F')
{
/* duree dépassée */
default=1;
Status = SendWithPrio ( cSurveilBal,(DoubleWord)&default, cMsgUrgent);
Status = Receive (cChienGardBal,(DoubleWord far *)&prise, cInfiniteWaiting);
}
default=0;
Status = SendWithPrio ( cSurveilBal,(DoubleWord)&default, cMsgUrgent);
}
return;
}

```

*figure 3.11. Extrait du programme de prise d'image*

### 7.3.4. Fonction "segmenter + interpréter image"

Cette activité est réalisée par un programme exécutable; elle ne fait intervenir aucune partie opérative "mécanique", elle peut donc être effectuée par n'importe quelle "machine de traitement d'information" connectée au réseau ETHERNET. La solution idéale eut été d'affecter à cette tâche un ordinateur "surpuissant" exclusivement dédié au traitement d'information dans la cellule. Dans notre maquette, le traitement est réalisé comme un processus fils (sous OS/2) lancé par la station qui a demandé l'exécution de la prise d'image.

Ce traitement nécessite le chargement d'un fichier d'étalonnage, d'un ou plusieurs fichiers modèles et du fichier exécutable lui même.

Le niveau de filtrage est assuré par le système d'exploitation OS/2 de la machine qui exécute le traitement. Les comptes-rendus renvoyés par le système sont :

- compte rendu d'exécution,
- défauts : programme introuvable, erreur d'exécution (division par zéro...)

Les comptes-rendus d'exécution sont :

- programme normalement exécuté,
- fichier image inexistant (défaut de chargement...),
- fichier d'étalonnage inexistant,
- fichier(s) modèle(s) inexistant(s),

Les résultats de l'analyse d'image sont directement "écrits" dans la base de données de la cellule.

### **7.3.5. Fonction "contrôler l'opération de vision"**

#### **7.3.5.1. Les traitements de contrôle de l'opération de vision**

Cette activité est réalisée par le programme de pilotage (écrit en C sous OS/2) implanté dans la station. Ce programme de pilotage [MRT 91] a pour objectif de commander l'exécution d'une phase sur une machine en contrôlant le déroulement des sous-phases et l'enchaînement des opérations. Nous avons vu qu'une sous-phase de vision fait intervenir des équipements non dédiés à la vision (axe du CHARLY-ROBOT); l'intégration de ces différents équipements pour la réalisation d'une tâche unique est le rôle de la station, elle permet d'assurer la **validation technologique, opérationnelle et fonctionnelle du capteur de vision pris comme un tout (éclairage, caméra, traitement...)**.

Le fichier d'entrée/sortie généré par l'outil ORCHIS-GRILLE est utilisé pour le prototypage de l'enchaînement des opérations au niveau station, ce point sera traité au chapitre IV.

### 7.3.5.2. Les données de contrôle de l'opération de vision

Le programme de pilotage de la station a accès à l'ensemble des données concernant le déroulement d'une opération de vision (programmes, fichier d'étalonnage, numéro de caméra etc...). Nous cherchons à concevoir la partie du Système d'Information de la cellule destinée à permettre le stockage et la recherche de ces informations.

#### 7.3.5.2.1. MCD de Référence

Le MCD de Référence est (cf. figure 3.3., para 5.5., chap III) : "une opération de vision" .

#### 7.3.5.2.2. MCD partiel

Nous instancions le MCD de Référence à notre système de vision particulier, nous obtenons ainsi un MCD partiel (cf. figure 3.12.). Ce modèle ne concerne que les opérations de contrôle qualité sur **une seule pièce à la fois** (ou une entité d'une pièce). Une *occurrence* d'OPER VISION est en relation avec au plus une *occurrence* de MODELE VISUEL (dans le cas d'un contrôle d'identification sur le poste de brut, une opération de vision devrait utiliser plusieurs modèles visuels).

#### 7.3.5.2.3. MOD particulier

L'*individu* OPER VISION est intégré en tant que *sous-type* de l'*individu* OPER QUALIF qui lui même est un *sous-type* de OPERATION dans le MOD particulier de la cellule (cf. figure 3.13.). Le lien qui existait dans le MCD Partiel entre l'*individu* MODELE VISUEL et l'*individu* OBJET est remplacé dans le MOD de la cellule par la *relation* REALISE ENTITE entre OPERATION et ENTITE GEOM. Un modèle visuel correspond donc à une ou plusieurs entités géométriques.

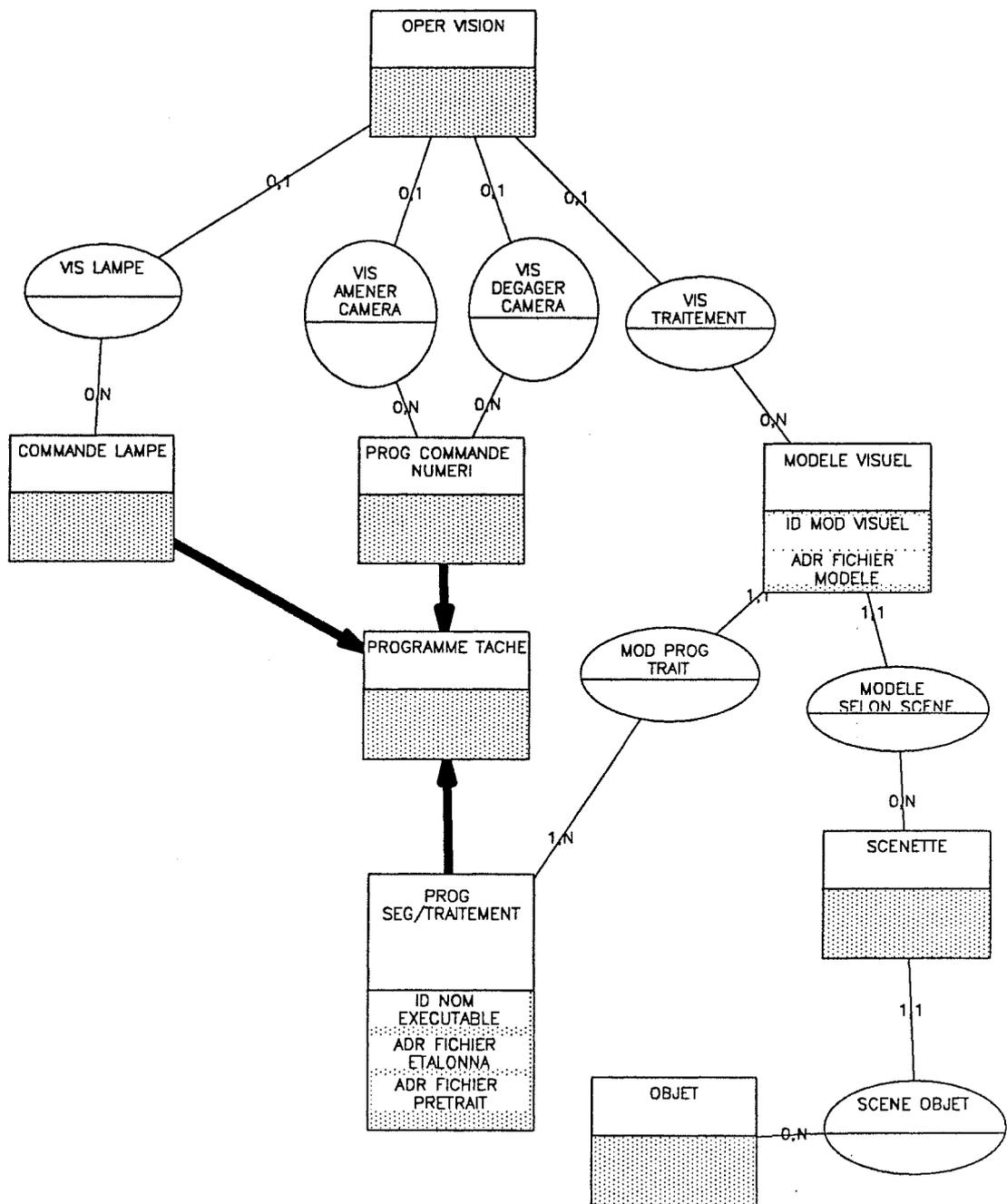
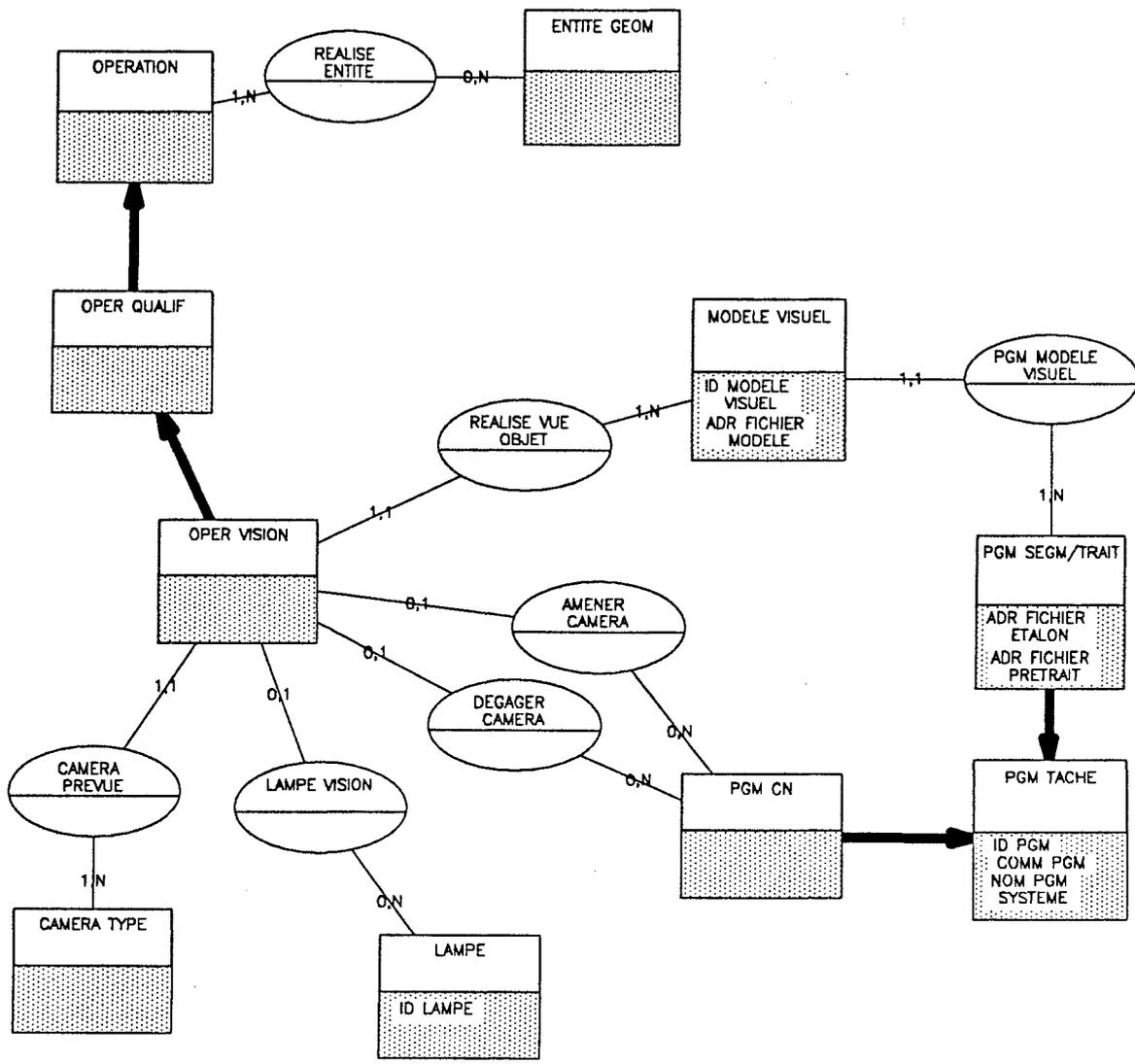


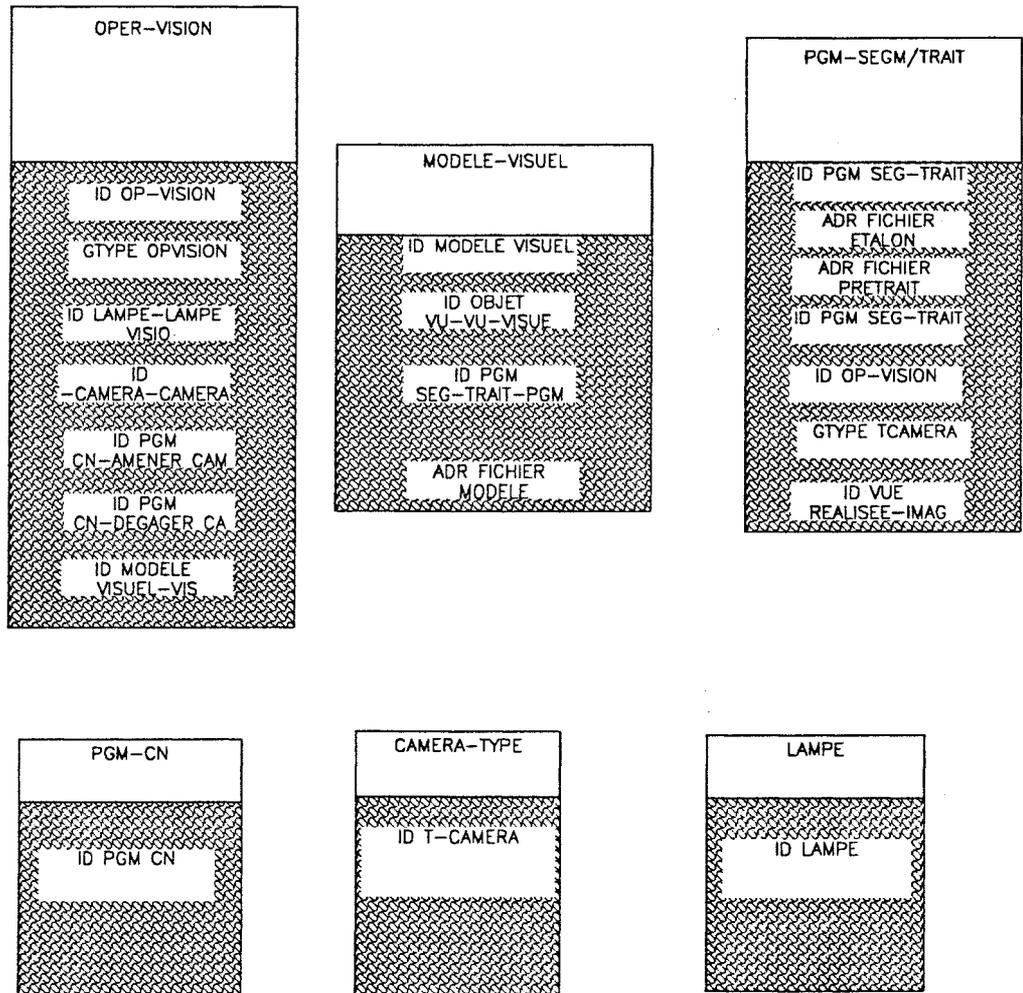
figure 3.12. MCD Partiel : opération de vision pour un contrôle d'une pièce



**figure 3.13. MOD Particulier : opération de vision dans la base de donnée de la cellule**

**7.3.5.2.4. MLD particulier et code SQL**

Le MLD et le code SQL sont directement dérivables du MOD particulier (cf. figure 3.14. et 3.15.).



*figure 3.14. MLD Particulier : opération de vision dans la base de données de la cellule*

```

use cellule
go
create table camera_type
    (id_t_camera smallint not null,
     gtype_tcamera varchar(20) null)
go
grant delete, insert, select, update
on camera_type
to ac_cfu
go
create table entite_geom
    (id_ent_geom smallint not null,
     gtype_ent_geom varchar(20) null,
     id_t_objet_geometrie smallint null,
     id_prim_geom_type_en smallint null,
     id_ent_geom_am_entit smallint null)
go
grant delete, insert, select, update
on entite_geom
to ac_cfu
go
create table lampe
    (id_lampe smallint not null)
go
grant delete, insert, select, update
on lampe
to ac_cfu
go
create table modele_visuel
    (id_modele_visuel smallint not null,
     adr_fichier_modele varchar(10) null,
     id_pgm_seg smallint null,
     id_objet_vu smallint null)
go
grant delete, insert, select, update
on modele_visuel
to ac_cfu
go
create table oper_qualif
    (gtype_opq varchar(20) null,
     stype_opq varchar(20) null,
     id_opq smallint not null,
     id_opq_opq_r_am smallint null,
     id_opq_opq_r_av smallint null,
     id_t_captteur_outil_o smallint null)
go
grant delete, insert, select, update
on oper_qualif
to ac_cfu
go
create table oper_vision
    (id_op_vision smallint not null,
     gtype_op_vision varchar(20) null,
     id_t_camera smallint null,
     id_modele_visuel smallint null)
go
grant delete, insert, select, update
on oper_vision
to ac_cfu
go
create table operation
    (id_oper smallint not null,
     stype_oper varchar(20) null,
     gtype_oper varchar(20) null)
go
grant delete, insert, select, update
on operation
to ac_cfu
go
create table pgm_cn
    (id_pgm_cn smallint not null)

```

*figure 3.15. Code SQL : opération de vision*

## Chapitre 4

# Intégration dans le système de gestion de production

1. Problématique .....	1
2. Concevoir les produits.....	2
3. Concevoir les processus.....	3
3.1. L'approche par variante (dérivation).....	4
3.2. L'approche générative .....	4
3.3. L'approche Généralisation/Particularisation .....	4
3.4. Situation des méthodes de conception des processus dans le "trièdre méthodologique" .....	5
3.5. Deux exemples de système de conception des processus .....	5
3.5.1. Le système expert PROPEL.....	5
3.5.2. Le projet Quick Turnaround Cell manufacturing - QTC.....	6
3.5.2.1. Modélisation du produit (design).....	6
3.5.2.2. Planification des processus (Process planning) [KAN 89].....	7
3.5.2.3. Inspection par vision (vision inspection) [PAR 88a,b] .....	7
3.6. Proposition d'une démarche de conception d'un MCD pour la conception des processus intégrant le procédé de vision.....	8
3.6.1. L'individu SPECIFS VISION.....	12
3.6.2. Les individus FORME et MATIERE .....	12
3.6.3. Les individus CONTROLE, PRECISION, CADENCE .....	13
3.6.4. L'individu ENVIRONNEMENT .....	14
4. Concevoir les moyens .....	14
4.1. Systèmes de conception des "connaissances" consommées par un système de vision	15
4.1.1. Quick turnaround cell manufacturing - QTC [PAR 88a][PAR 88b].....	15
4.1.1.1. Modélisation et interface .....	15
4.1.1.2. Préparation du traitement .....	16
4.1.2. CADG-Based Computer Vision [HAN 87].....	17
4.1.2.1. La représentation géométrique .....	17
4.1.2.2. La sélection automatique des paramètres de reconnaissance.....	17
4.1.2.3. La synthèse de l'arbre de stratégie .....	18
4.1.3. Conception automatisée de système de vision [KAT 87] [KAT 88] .....	19
4.1.3.1. Modélisation d'une attitude de pièce.....	19
4.1.3.2. Modélisation des senseurs (capteurs).....	20
4.1.4. 3D Part Orientation System [BOL 86] .....	20
4.1.4.1. Modélisation de l'objet .....	20
4.1.4.2. Sélection des primitives.....	21
4.1.5. Knowledge-Based-Robotics [KAK 87] .....	22
4.1.5.1. Représentation spéciale capteur (sensor-tuned-representation)..	22

4.1.5.2. Segmentation d'image 3D et comparaison .....	23
4.1.6. Conclusion sur les systèmes de conception des "connaissances" consommées par un système de vision et proposition d'un MCD .....	23
4.2. Un cycle de conception d'un processus de contrôle par vision .....	26
4.2.1. Le besoin en génie visionique.....	26
4.2.1.1. La notion de cycle de vie.....	27
4.2.1.2. La réutilisabilité .....	28
4.2.1.3. Les objets .....	28
4.2.2. Application à la visionique .....	28
4.2.2.1. La phase de spécification .....	29
4.2.2.2. La phase d'analyse.....	29
4.2.2.3. La phase de conception .....	30
4.2.2.4. La phase de codage .....	30
5. Notre approche pour le prototypage du contrôle-commande de niveau station intégrant le procédé de vision .....	30
5.1. Prototypage du contrôle par vision des perçages de la pièce NASA sur le CHARLY-ROBOT .....	31
5.1.1. Génération de la gamme d'usinage par PROPEL.....	31
5.1.2. Insertion des opérations de contrôle dans la sous-phase .....	33
5.1.3. Prototypage sur SPEX.....	35
5.1.3.1. Programmation Fonctionnelle.....	35
5.1.3.2. Fusion des actinomies .....	37
5.1.3.3. Utilisation sur SPEX .....	40
5.1.3.3.1. L'application .....	40
5.1.3.3.1.1. Le niveau équipement (environnement) ...	40
5.1.3.3.1.2. le niveau station (système).....	41
5.1.3.3.2. Prototypage du contrôle-commande de la station .....	41
5.1.3.3.2.1. Démarche .....	41
5.1.3.3.2.2. Boîte Fonctionnelle "CONTROLE PAR VISION" .....	43
5.1.3.3.2.3. Boîte Fonctionnelle "USINAGE" .....	43
5.1.3.3.2.4. Boîte Fonctionnelle des autres procédés...43	
5.1.3.3.2.5. Fusion des Boîtes Fonctionnelles .....	46
5.1.3.3.2.6. Boîte Fonctionnelle "SOUS-PHASE" .....	46
5.2. Conclusion sur la méthode de prototypage .....	47

## Chapitre 4

# Intégration dans le système de gestion de production

### 1. Problématique

L'intégration d'une nouvelle technologie telle que la visionique dans un environnement de production intégré concerne non seulement les activités de production proprement dites mais également les activités dites "hors-ligne" d'organisation de la production. Ces activités sont la *conception, l'achat, l'approvisionnement, la planification, la gestion des ressources physiques et humaines, la maintenance et la qualité*. Ces activités sont dites "horizontales", l'ensemble constitue le système de *gestion de production*.

Nous nous sommes limités dans notre étude à la décomposition "verticale" de l'activité *conception*.

La recherche dans ce domaine concerne aussi bien les objectifs à atteindre (la conception) que la manière d'atteindre ces objectifs (les méthodes de conception);

Nous étudions successivement :

- la *conception des produits* : où le système de vision exerce une rétroaction sur la définition du produit,
- la *conception des processus* : où l'intégration du processus de vision nécessite la création d'une expertise du procédé de vision; nous proposons une démarche de conception d'un MCD pour le choix des moyens de vision,
- la *conception des moyens* : où la plupart des travaux de recherche se limitent à la conception des programmes de vision; nous proposons un MCD pour la conception des programmes en vision; Nous présenterons ensuite un cycle de

vie de conception d'une application de vision [VOG 91] qui constitue un premier pas en Génie Visionique.

Après avoir montré le manque d'outils et de méthodes de conception intégrés adaptés à la vision, nous apportons une contribution au "Génie Visionique" en proposant *une méthode de prototypage des processus opératoires intégrant le procédé de vision*.

## 2. Concevoir les produits

Dans un contexte de CIM, l'intégration des diverses fonctions de l'entreprise (conception, planification et fabrication) impose un lien plus étroit entre ces différents secteurs d'activité. Ce lien s'établit à travers le partage des données de l'entreprise. Lorsque qu'un industriel décide de lancer la fabrication d'un nouveau produit, son souci majeur est de s'assurer de sa **manufacturabilité**; c'est à dire : sa capacité à fabriquer ce nouveau produit. Ainsi dès la conception du produit, on s'intéresse à sa fabrication. La **rétro-action qu'exerce les moyens de fabrication sur la conception** ne doivent en aucun cas remettre en cause la fonctionnalité du produit conçu. La prise en compte des moyens de fabrication dès la conception permet simplement de concevoir des produits qu'on est sûr de pouvoir fabriquer [CHA 89], et ce à moindre coût. On peut par exemple éviter de se doter de moyens d'usinage ou de manutention trop spécifiques à un type de produit, donc trop coûteux.

Un capteur de vision peut simplifier un processus de fabrication, l'emploi d'un système de vision peut alors entraîner une modification de la forme d'un produit si cette modification n'altère pas les fonctionnalités du produit et si elle permet l'emploi de la visionique.

Chaque élément du système de fabrication (usinage, manutention, contrôle) entretient des liens particuliers de **manufacturabilité** avec des produits à réaliser. Une nouvelle approche de conception des produits est née qui succède aux CAO classiques et qui utilise la notion de "design feature" : entité de définition correspondant à des opérations d'usinage.

Le système de vision en tant qu'élément du système de fabrication n'échappe pas à cette règle; lors de l'étude du Modèle Conceptuel de Données pour le choix des moyens en vision, nous montrerons que le système de vision **doit établir des liens de détectabilité** avec le produit (ici le mot détectabilité est la particularisation du mot manufacturabilité au cas d'un système de vision).

### 3. Concevoir les processus

Un processus de fabrication (dans le projet MRT) comprend essentiellement trois processus de base :

- le processus de transformation (Transformation),
- le processus de contrôle qualité, (Qualité)
- le processus de manutention (Manutention).

La conception du processus de fabrication a pour objet l'organisation des processus de base en fonction des ressources d'usinage, de contrôle et de manutention disponibles. Il s'agit de générer une gamme de fabrication, basée sur une gamme d'usinage et complétée par les opérations de contrôle et de manutention. L'objectif est d'automatiser la conception du processus de fabrication depuis la définition du produit jusqu'à la génération du code si possible.

Les méthodes de conception sont incluses dans un trièdre :

- *la dérivation* : dériver des solutions nouvelles à partir de solutions existantes de même niveau se rapportant à des problèmes similaires,

- *la génération* : générer une solution entièrement nouvelle en réutilisant uniquement des raisonnements ou des connaissances; les systèmes génératifs "possèdent" le "savoir-faire" du domaine auquel ils se rapportent

- *la Généralisation/Particularisation* : élaborer des solutions particulières à partir de solutions plus générales (modèles de référence).

### **3.1. L'approche par variante (dérivation)**

L'approche par variante [NAD 91], la plus courante, est à la base des progiciels de TGAO (techniques de groupe). Elle a pour objet une réutilisation optimale de l'expérience de l'entreprise, avec pour principe de dériver des solutions élaborées lors de la production de pièces antécédentes ressemblantes; un codage numérique permet de représenter des caractéristiques aussi diverses que matière, morphologie, fonctionnalité,... Cette approche a fait ses preuves dans des secteurs d'activités relativement peu diversifiés où la notion de dérivation s'applique logiquement; par contre ces techniques de groupe (trop pointues) sont peu compatibles avec la notion de flexibilité et la diversité des technologies mise en cause dans l'utilisation d'un îlot automatisé de production (centre d'usinage, machine à mesurer, système de vision, robots, etc...).

### **3.2. L'approche générative**

L'approche générative propose de concevoir une gamme d'usinage à partir [TSA 87][ANS 91]:

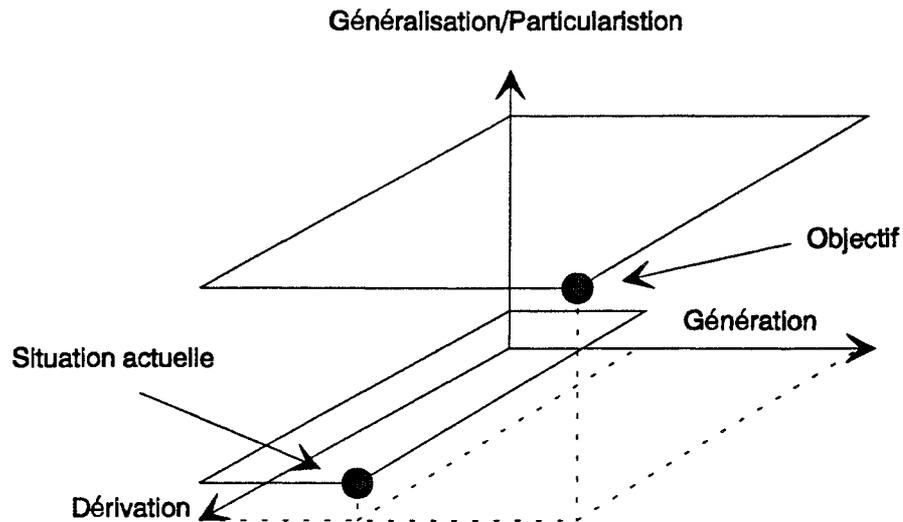
- de règles de l'art propres au procédé de transformation,
- d'une description des ressources de l'atelier,
- d'une description morphologique du produit.

### **3.3. L'approche Généralisation/Particularisation**

Cette approche n'est pas utilisée pour concevoir la gamme d'usinage proprement dite, mais elle lui succède dans les phases ultérieures de conception plus fine. Nous montrerons au paragraphe 5. de ce chapitre comment cette approche peut permettre de prototyper les processus opératoires de manière plus fine par instanciation de "comportements génériques de processus de base (TQM)" après avoir utilisé une approche générative avec l'outil PROPEL.

### 3.4. Situation des méthodes de conception des processus dans le "trièdre méthodologique"

Les méthodes de conception des processus se situent donc surtout sur l'axe dérivation, moyennement sur l'axe génération et quasiment pas sur l'axe Généralisation/Particularisation (cf. figure 4.1.). Notre objectif est d'exploiter ce dernier dans les phases de conception détaillée des processus opératoires (cf. para 5.).



*figure 4.1. Situation des méthodes de conception des processus dans le trièdre méthodologique*

### 3.5. Deux exemples de système de conception des processus

#### 3.5.1. Le système expert PROPEL

Le système expert PROPEL (approche générative) [TSA 87] utilise une description par entité du produit et propose pour une pièce donnée et pour des ressources données toutes les gammes possibles de fabrication. Les gammes proposées par PROPEL se composent d'une séquence hiérarchique de phases (sur une même machine), de sous-phases (même appui), d'opérations (même outil) et d'usinages (ébauche, finition ...). Ces gammes ne travaillent que sur des "types de machine" (le système ne connaît pas de machine particulière).

PROPEL n'intègre pas les opérations de contrôle et de manutention. Les différents niveaux de décision sont :

- niveau 1 : qu'usiner? Il s'agit de décrire la géométrie du produit à usiner avec un langage à base d'entité du gammiste;
- niveau 2 : dans quel ordre et sur quel type de machine faut-il réaliser les opérations d'usinage? Pour résoudre ce problème on utilise un ensemble de règles expertes propres au procédé d'usinage qui infèrent à partir de la définition du produit (par entité) et des ressources disponibles de l'atelier pour calculer toutes les gammes d'usinage possibles.

### **3.5.2. Le projet Quick Turnaround Cell manufacturing - QTC**

L'objectif du projet QTC [CHA 89] (quick turnaroud cell : fabrication par contournage) est d'intégrer la conception (design), les processus (process planning), le contrôle de cellule (cell control) et les opérations de contrôle (inspection function) dans un système de fabrication capable de produire des pièces rapidement en utilisant un minimum de connaissances et de main d'oeuvre humaine. Dans un tel environnement, Chang définit le rôle de la vision : "*Are the correct tools mounted in the machine? Did the operator fixture the stock correctly? Is machining process proceedings normally? Is the final part within specification?*"

#### **3.5.2.1. Modélisation du produit (design)**

Selon [CHA 89], deux approches sont envisageables, l'une est d'utiliser une CAO classique et de développer une interface spéciale pour reconnaître les éléments de fabrication à partir du modèle; son avantage est de permettre l'utilisation d'une CAO quelconque, le gros désavantage est le problème de reconnaissance de critère de fabrication permettant de faire le lien avec les moyens de fabrication. La seconde approche consiste à concevoir le produit directement en terme d'entités reconnaissables pour la planification de la fabrication. Ces entités sont donc spécifiques aux procédés de fabrication. Un tel environnement limite le concepteur à définir des produits forcément manufacturables. La deuxième approche semble la plus adaptée à une intégration

réussie des différents procédés, même si elle favorise par définition le procédé d'usinage. Elle nécessite l'élaboration d'une expertise définissant les conditions de **manufacturabilité** d'un ensemble "d'entité de fabrication" d'un point de vue contrôle et visionique.

### **3.5.2.2. Planification des processus (Process planning) [KAN 89]**

Le projet QTC utilise un système appelé AMPS (Automatic Machining Planning System). Ce système repose sur quatre modules :

- choix des processus,
- choix des outils,
- choix des montages d'usinage,
- enchaînement des opérations,
- génération du code.

AMPS utilise quatre bases de données :

- base de données processus,
- base de données montages,
- base de données outils,
- base de données machines.

Le système génère une gamme (process plan) qui sert d'interface entre la conception des processus et l'exécution des processus (Cell Control).

### **3.5.2.3. Inspection par vision (vision inspection) [PAR 88a,b]**

Les opérations de contrôle par vision sont intégrées dans la gamme d'usinage après la génération du "process plan". Le système QTC intègre une opération de contrôle d'outil systématiquement après chaque changement d'outil dans la gamme et un contrôle du produit après chaque opération d'usinage et chaque changement de montage. Il n'y a donc pas d'expertise proprement dite pour le choix du lieu d'insertion des contrôles; la gamme d'usinage sert de référence absolue pour l'ensemble du processus de transformation du produit.

### 3.6. Proposition d'une démarche de conception d'un MCD pour la conception des processus intégrant le procédé de vision

Intégrer le procédé de vision dans une gamme d'usinage, c'est répondre dans cet ordre aux deux questions suivantes : que contrôler? Avec quoi et comment? Lorsque l'on conçoit les processus intégrant le contrôle, on répond intégralement à la première question, et en partie à la seconde. En partie seulement parce que il est inutile de détailler le "comment" à ce niveau; **pourtant, il impératif de maîtriser la sémantique ("comment") du procédé impliquant le "quoi" pour être certain de la validité du choix effectué (c'est à dire : être certain que le contrôle est possible avec le matériel choisi)**. La notion de **manufacturabilité** [CHA 89] est la clef de la réalisation de la conception des processus. Le mot manufacturabilité signifie pour une entité de définition du produit qu'il existe un lien entre l'entité de définition du produit et les outils qui la réalisent. Si ce lien existe entre entité et outil, alors on peut dire que cette entité est manufacturable. Ce lien existe toujours pour le procédé d'usinage, puisque les entités utilisées actuellement dans les systèmes de "design feature fabrication" ont été créés pour le procédé d'usinage. Qu'en est-il des autres procédés? L'équivalent du mot manufacturabilité peut être, pour un système destiné à capter de l'information (capteur tout ou rien, capteur de pression, de proximité, palpeur, système de vision), le mot **déteçtabilité**. Pour le procédé de vision, la déteçtabilité signifie : l'entité peut-elle être "vue" ou "contrôlée par un système de vision? Il apparaît évident que les liens implicites liant "entité de définition" et "opération d'usinage" sont inopérants pour la vision, une opération de vision n'a rien en commun avec une opération d'usinage.

Le problème se pose de la manière suivante : sachant quelle est la forme de l'objet à "traiter par vision", sachant quel type de traitement on désire réaliser (identification, localisation, mesure dimensionnelle, test de présence, vérification d'état de surface...), sachant quelle est sa matière (aluminium, bois, acier, ...), sachant à quelle précision on veut travailler, sachant combien de temps on dispose pour réaliser le traitement (temps masqué, temps réel), existe-t-il un système de vision connu (et existant dans l'atelier si possible) capable de réaliser cette **opération de vision**? Celui ci sera alors défini par : l'optique, l'éclairage, le capteur, la carte de vision, le prétraitement, le type de segmentation, et le traitement (Bayes, prédiction-vérification, plus proche voisin ...). **Il s'agit donc**

**d'établir une relation entre un ensemble de spécifications de la tâche de vision et un ensemble d'éléments de vision cohérent et apte à réaliser les spécifications.**

La résolution de ce problème passe par deux points :

- la *connaissance des systèmes de vision* : nous avons énuméré dans le chapitre II les informations qui composent un système de vision (*individus* MONTAGE OPTIQUE, MONTAGE TRAITEMENT, MONTAGE SOURCE, MONTAGE CAPTEUR); l'ensemble de ces informations prises ainsi constitue un "catalogue de vision"; pourtant **il existe des relations sémantiques entre les individus du système de vision qui constituent le savoir-faire de visionique** (exemple de relation : une extraction de surface se fait sur une image 2D, un algorithme utilisant la méthode de Bayes fonctionne avec des primitives visuelles globales, les primitives visuelles globales peuvent être extraites à partir d'image 2D et 3D, etc...); ces relations caractérisent la sémantique du domaine vision.

**L'objectif à atteindre est la formalisation de cette sémantique de façon à aller au delà de la création d'un simple catalogue d'éléments de vision;** par la coopération entre les métiers de l'optique, de l'électronique des capteurs, de l'éclairage et du traitement d'image, il faut établir l'ensemble des *relations* et des *contraintes* permettant de modéliser cette sémantique; la figure 4.2 en montre une partie à travers la représentation des traitements en vision et de leurs relations avec les autres éléments d'un système de vision; **la connaissance de cette sémantique est indispensable au choix des moyens** (sans quoi on pourrait choisir de monter une caméra linéaire avec une carte de numérisation matricielle et avec des traitement de télémétrie laser!);

- le *lien avec les spécifications* : nous sommes amenés à utiliser pour la vision la même technique que celle employée par les usineurs : créer des **liens entre outils de vision cohérents et produit à observer.**

Pour créer ces liens complexes existant entre des systèmes de vision et des spécifications de contrôle par vision, il faut modéliser une structure de données (*individu* SPECIF VISION) à l'image des méthodes couramment employées en vision pour le choix des systèmes.



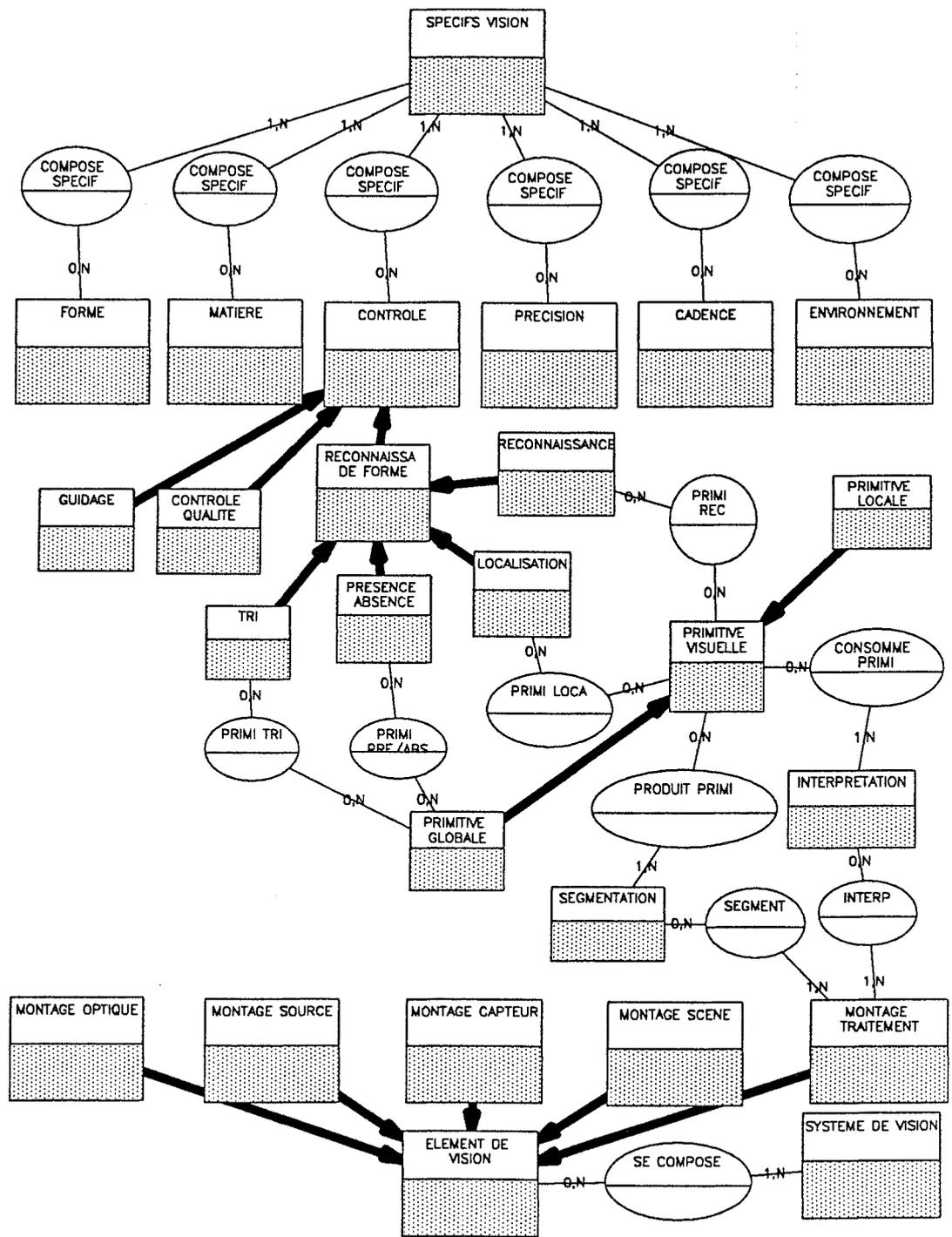


figure 4.3. Extrait du MCD de vision : le choix des moyens

Cette structure de données permet de définir, connaissant la définition d'une application (forme, matière, type de contrôle, cadence, environnement,...), quels sont les types de système de vision (primitives visuelles correspondantes, traitements possibles, matériels ...) (cf. figure 4.3.) aptes à réaliser les spécifications.

La création de ces liens revient à formaliser le **savoir-faire de choix des moyens en vision**.

Ainsi, pour chaque *individu* du système de vision, on doit savoir avec quoi il peut fonctionner, et pour quel type de spécification.

### 3.6.1. L'*individu* SPECIFS VISION

L'*individu* SPECIFS VISION matérialise le concept de *spécifications du système de vision*, il est défini par des relations avec ces différentes composantes (forme, matière, type de contrôle, précision, cadence, environnement) [MEZ 88][DUP 88].

### 3.6.2. Les *individus* FORME et MATIERE

Ces *individus* caractérisent entièrement l'objet sur lequel doit porter le traitement de vision. Cet objet, d'un point de vue vision, est un objet réel intégré dans un environnement, qu'il soit outil ou produit, cela n'a pas d'importance pour le traitement de vision qui est effectué. A ce niveau de conception, aucune information géométrique précise n'est consommée. Voici les notions que recouvre le concept de "forme" :

- dimensions de l'objet à regarder (hauteur, longueur, largeur), les dimensions de l'objet permettent de définir la résolution de la caméra et du numériseur ainsi que l'optique
- distance possible entre objet et caméra (détermination de la distance focale et du zoom)
- dimensions de la scène (détermination du champ image de l'optique).

L'information de MATIERE vient compléter celle de FORME :

- matière et couleur de l'objet (détermination du nombre de niveaux de gris)
- matière et couleur de la scène (détermination de l'éclairage).

### 3.6.3. Les *individus* CONTROLE, PRECISION, CADENCE

Ces trois *individus* décrivent l'application à réaliser dans le cahier des charges.

Contrôle :

- reconnaissance de forme :
  - tri de pièces
  - contrôle de présence/absence
  - reconnaissance de pièces
  - orientation de pièces
  - localisation de pièces
  - comptage de pièces
  - reconnaissance de caractères
  - autres
- contrôle de qualité :
  - état de surface
  - contrôle de cote
  - recherche de défauts
  - contrôle d'impression
  - autres
- guidage :
  - suivi de trajectoires
  - guidage de robot
  - guidage de machine
  - autres.

Précision :

- tolérance de la mesure

- précision de la mesure (détermination des caractéristiques de l'élément photosensible).

Cadence :

- vitesse de déplacement
- distance entre deux objets
- nombre d'objets à traiter
- temps disponible entre deux lots
- nombre de postes de travail.

### **3.6.4. L'*individu* ENVIRONNEMENT**

Cet *individu* sert à caractériser l'environnement de travail dans lequel doit se dérouler le traitement. Le concept d'environnement recouvre les propriétés suivantes :

- propreté : poussière, vapeur, huile, graisse, hygrométrie (besoin d'isolation)
- température
- humidité
- éclairage ambiant : naturel, électrique, éclairage constant indépendant du jour
- parasitage
- volume d'implantation
- environnement mécanique
- environnement humain (nuisances visuelles).

## **4. Concevoir les moyens**

La conception des moyens de contrôle définit les moyens techniques et logiciels capables d'accomplir une opération de contrôle planifiée lors de la conception des processus. Nous présentons tout d'abord des travaux pour lesquels la conception se limite à la réalisation de programmes de vision adaptés à des systèmes de vision prédéfinis et fonctionnant avec des traitements figés. Nous montrerons que ces travaux doivent s'intégrer dans une démarche plus générale

de conception incluant les notions héritées du Génie Automatique de "cycle de vie".

#### **4.1. Systèmes de conception des "connaissances" consommées par un système de vision**

Un certain nombre de travaux ont été effectués sur la génération automatique de programmes de reconnaissance d'objet à partir d'une définition géométrique de l'objet en 3D réalisée sur un système de CAO (classique) et d'une modélisation des senseurs utilisés. L'ensemble des travaux que nous allons présenter s'appliquent à des traitements d'image prédéfinis : il n'y a pas de problème de choix de caméra, d'éclairage, et même d'algorithmes. Ces travaux exposent donc des méthodes de conception des "données" consommées par les algorithmes de traitement lors de leur exécution. Ces données sont le plus souvent des modèles d'objet ou des arbres de stratégie.

##### **4.1.1. Quick turnaroud cell manufacturing - QTC [PAR 88a][PAR 88b]**

Le système de Park utilise des images 2D pour interpréter des pièces de dimension trois, son travail ne considère que le cas de pièces isolées et entièrement visibles sur un fond uniforme.

###### **4.1.1.1. Modélisation et interface**

*" A design feature based representation scheme is the newest approach to achieve the interface between CAD and CAM technology".* Une CAO classique utilise des entités géométriques de type cylindre, parallélépipède et ellipsoïde qui ne contiennent que des informations géométriques (d'aspect); une CAO par entité fournit des informations supplémentaires qui seront utiles à la fois au concepteur et à la fois à la mise en place de la production. Les entités utilisées par le système de Park sont basées sur les opérations d'usinage (machining operations). Ces entités ne sont pas seulement des informations géométriques, elles sont autant de procédés de fabrication. Une entité est décrite par ses attributs géométriques, sa position relativement à un repère de référence et sa taille. **Chaque entité correspond à une opération d'usinage.** Les informations de tolérances peuvent être facilement attachées aux entités.

"..., it is desirable to provide a solid modeler from which vision programs can extract geometric information in addition to the high level design system., Hence, a 3D boundary representation model is generated from the high level feature based description." A partir de cette représentation de haut niveau de la pièce en terme d'entités d'usinage, Park fabrique une représentation intermédiaire (boundary representation) qui est le point d'ancrage du paramétrage des algorithmes de vision. Cette représentation décrit la pièce en terme de contours, droites, faces, on se rapproche du monde visuel. *"this capability now allows us to use both the design feature-based representation (which is best for design and tolerance specification) and the boundary representation solid model (which is best for vision feature understanding)."* Nous retrouvons ici le concept de représentation du monde réel et de représentation du monde visuel (cf. chap II). Park utilise une machine capable d'établir les liens entre les deux hiérarchies réelle et visuelle. Son système est capable de générer toutes les représentations 3D des pièces intermédiaires par adjonction d'entités successives (ou d'opérations d'usinage)

#### 4.1.1.2. Préparation du traitement

La préparation de la reconnaissance se fait en cinq étapes dont la plus importante consiste à fabriquer à partir du "boundary model" les soixante vues de la pièce (viewing sphere); ce travail est énorme, mais pas insurmontable, cela n'est rien d'autre que de la fabrication d'image par ordinateur, les systèmes de CAO performants le font très bien. Cette étape est l'antinomie parfaite du processus de compréhension d'image où l'information est restructurée, ici au contraire il y a éclatement de la structure, l'information stockée (rangée sous forme 3D) est dispersée en soixante vues 2D de la pièce, il n'y a pas de perte d'information (le processus est réversible), il y a perte de structure. On fait en quelque sorte le chemin en sens inverse. Les modèles 2D obtenus font le lien entre un objet 3D et ses différentes attitudes (position), chaque modèle correspond à une position de l'objet par rapport à l'observateur. Ces modèles sont composés d'éléments tels : droites, arcs, ellipses, régions adjacentes... La deuxième étape consiste à calculer pour chaque vue le descripteur de Fourier à partir des contours extérieurs de l'objet, la troisième étape sert à choisir les critères de sélection les plus discriminants (lignes longues, trous ...), on calcule

ensuite les relations entre ces primitives (angles, place, distance, ...). Dans l'étape quatre on établit la liste de primitives visibles (qui formeront le groupe d'indices de vérification) et finalement l'étape cinq sert à déterminer quels seront les indices visuels qui serviront de base à la localisation précise de l'objet (deux droites parallèles).

#### **4.1.2. CADG-Based Computer Vision [HAN 87]**

Hansen cherche à établir le lien entre CAO et Vision par ordinateur. Une méthode de génération automatique de stratégie de reconnaissance basée sur les propriétés géométriques des formes est discutée et testée. Pour établir ce lien entre vision et CAO, Hansen doit résoudre trois problèmes : la représentation géométrique (geometric knowledge representation), la sélection automatique de critères (automatic feature selection) et la synthèse de l'arbre de stratégie (strategy tree synthesis).

##### **4.1.2.1. La représentation géométrique**

Un modèle spécifique (Alpha\_1 B\_spline) est utilisé pour modéliser des surfaces quelconques. Pour obtenir les entités utiles à la reconnaissance en 3D, des techniques de transformation en une représentation pour la vision par ordinateur ont été développées. Les relations entre entités doivent être explicites, Hansen insiste sur le fait que le système de génération automatique doit pouvoir avoir facilement accès aux données géométriques du modèle. On utilise donc une représentation intermédiaire spécifique aux besoins de la vision.

##### **4.1.2.2. La sélection automatique des paramètres de reconnaissance**

La représentation géométrique de l'objet permet la sélection d'un ensemble complet et cohérent de critères, ainsi que la connaissance que l'on a du système de vision utilisé (informations sur la sûreté et la robustesse des moyens d'extraction de ces critères). Le système sélectionne les paramètres de reconnaissance selon plusieurs critères :

- faible fréquence d'apparition,
- facilité de mesure,

- complexité de la mesure,
- suffisance, l'ensemble de paramètres couvre-t-il toutes les vues de l'objet?
- discrimination du paramètre.

Il y a deux types de crédibilité d'un paramètre que le système peut quantifier, la crédibilité du paramètre lui même et la crédibilité de son mode d'extraction. A partir du modèle 3D, le système génère les différents aspects de l'objet (tessellation sphere), puis il choisit quelles seront les primitives visuelles (paramètres, droites) qui seront les plus aptes à engendrer une reconnaissance.

#### **4.1.2.3. La synthèse de l'arbre de stratégie**

Le paradigme de reconnaissance appliqué par Hansen est le paradigme dit de "prédiction vérification". La phase de génération d'hypothèse est sous contrôle de l'arbre de stratégie et la phase de vérification tente d'affirmer ou d'infirmer les prédictions de la première phase.

Un arbre de stratégie se compose de trois parties :

- la base (root) qui représente l'objet lui même
- les paramètres (indépendant des vues) de niveaux 1 qui permettent une identification et une localisation grossière de l'objet
- l'arbre secondaire de vérification des hypothèses faites au niveau 1 (CES, Corroborating Evidence Subtrees).

L'arbre de stratégie joue le rôle de modèle d'objet, la manière de reconnaître et de localiser celui-ci et sa définition géométrique étant intimement mêlés dans la syntaxe de l'arbre. La reconnaissance se solde par l'établissement de liens entre des paramètres ou primitives visuelles apprises avant et des paramètres extraits de la scène. La conduite la plus sûre de fabrication des liens étant contenue dans l'arbre de stratégie.

Il est important de retenir que Hansen fait intervenir pour la programmation de son système de reconnaissance des données géométriques mais aussi des données sur le procédé de vision (algorithmes, robustesse, précision, matériel...); cette remarque est importante car elle implique l'existence de liens entre données géométriques et traitements de celles-ci par le système de vision. Elle nécessite donc une qualification (une description) du système de vision en tant qu'outil

capable d'effectuer des opérations caractérisées par certains critères (algorithmes, robustesse, précision, matériel...) sur un produit lui même caractérisé par sa géométrie.

#### **4.1.3. Conception automatisée de système de vision [KAT 87] [KAT 88]**

Katsushi Ikeuchi décrit une méthode de génération d'algorithmes de reconnaissance d'objet en 3D à partir d'un modèle géométrique en vue de leur localisation pour une manipulation robotique. D'après Katsushi, la modélisation d'objet, la modélisation des capteurs, la prédiction d'apparence, la génération de stratégie et la génération de programme sont les problèmes inévitables de la conception automatisée de système de vision

La procédure est quasiment la même que précédemment : ayant comme point d'entrée un modèle 3D de l'objet (CAD model), le système génère tout d'abord les soixante apparences de l'objet vu selon soixante directions différentes. Ces apparences sont classifiées en groupes (aspect, attitude) basés sur une face dominante ou un critère dominant. A partir de cette classification primaire des apparences 2D de l'objet, le système génère l'algorithme de reconnaissance qui a la forme d'un arbre d'interprétation. L'arbre d'interprétation se décompose en deux fonctionnalités : une première qui détermine l'attitude globale (ou groupe d'aspect) de l'objet, et la deuxième qui complète la première en précisant la position de l'objet dans une attitude données.

##### **4.1.3.1. Modélisation d'une attitude de pièce**

Chaque attitude est définie selon ces caractéristiques :

- moment d'inertie des faces
- relation entre faces (position relative)
- profil angulaire de la face (distance entre centre de masse et contour en fonction de l'angle)
- relation entre bords (droites)
- image gaussienne étendue (EGI), histogramme de la distribution des orientations de l'objet (critère global)
- distribution des caractéristiques des surfaces (cylindre, plan, ellipse, hyperbole).

L'arbre d'interprétation est construit de façon à utiliser au mieux la discrimination de celles-ci entre elles; le paradigme de prédiction vérification est appliqué selon les directives de l'arbre. Les critères globaux (inertie, EGI) sont d'abord utilisés pour déterminer le groupe d'attitude, puis les critères locaux (droites, surfaces) sont consommés afin de parachever la localisation. La génération de l'arbre de décision se fait à partir d'un ensemble de règles, ces règles sont générées par l'expert vision.

#### **4.1.3.2. Modélisation des senseurs (capteurs)**

La modélisation des capteurs a pour but de déterminer la détectabilité d'une forme (d'un critère, d'une primitive visuelle) et la précision de mesure de celle-ci. Ces deux caractéristiques dépendent de la position du capteur par rapport à la scène, c'est pourquoi Katsushi définit un "espace de configuration d'une forme" par rapport à un senseur dans lequel il fait intervenir une description de la scène (distance entre caméra et forme, position de la forme, reflets, transparence de l'aire ambiant, intensité de la lumière, direction de la lumière...). Cette modélisation très complexe permet de définir les contraintes de prise de vue.

#### **4.1.4. 3D Part Orientation System [BOL 86]**

Bolles et Horaud présente un système qui identifie des objets, les vérifie et les localise. L'objectif recherché est la génération automatique de stratégie de traitement basée sur une définition 3D de l'objet et sur la géométrie du senseur. Tout le système est basé sur le principe de prédiction vérification.

##### **4.1.4.1. Modélisation de l'objet**

*"Computer-aided-design (CAD) system and their representation are intended for constructing and displaying objects, not recognising them. Even though a CAD model may be complete in the sense that it contains all the 3D information about an object, there are more convenient representations for a recognition system."* Selon Bolles, la représentation initiale de l'objet en CAO est nécessaire

mais pas suffisante pour intégrer l'automatisation de la conception d'un système de vision. Bolles utilise quatre modèles :

- le modèle CAO, "volume, surface, edge, vertex description and pointers",
- le réseau de classification des primitives visuelles qui sert à diriger le traitement "en ligne", à chaque primitive visuelle correspond une procédure d'extraction,
- la représentation par petits éléments plans (planar patch description), est utilisée pour prédire rapidement l'orientation d'une surface,
- le modèle filaire qui est une liste de primitives (cylindre, plan) et qui sert à l'affichage des hypothèses.

Les trois derniers modèles sont déduits du premier. Bolles fait la distinction entre les relations indépendantes des vues (view-independent relationships) qui sont les relations spécifiées directement dans le modèle 3D et les relations dépendantes des vues (view-dependent relationships) qui sont le fruit du couplage de la position du capteur par rapport à la scène et son type.

#### 4.1.4.2. Sélection des primitives

La première primitive (critère visuel) que le système extrait doit être la plus discriminante et la moins coûteuse. Le choix des autres primitives se fait selon des critères tels que : la réduction du nombre d'interprétation, la vérification d'une hypothèse, la détermination de degrés de liberté... Les primitives utilisées par le système de Bolles sont les bords (intersection de surface); il définit trois principaux types (straight dihedral, circular dihedral et straight tangential) caractérisés par des propriétés (longueur, profondeur, rayon...). Il faut remarquer que les primitives utilisées par Bolles sont définies dans un espace à trois dimensions et non deux, c'est pourquoi il parle de bords et non de droites. Cela suppose la possibilité d'extraire ces primitives directement en 3D; en effet Bolles utilise un "white scanner" qui permet de travailler en trois dimensions par triangulation.

La phase "en ligne" est classique, le système fait une acquisition de données (image 3D laser), les primitives sont extraites dans un ordre calculé par l'arbre de décision, une hypothèse est faite, qui sera vérifiée par la suite. Il faut remarquer qu'à aucun moment le système de Bolles ne se sert d'une modélisation 2D de l'objet (aspect).

#### 4.1.5. Knowledge-Based-Robotics [KAK 87]

Kak propose une représentation intermédiaire (sensor-tuned-representation) des objets à manipuler en vision destinée à faciliter le lien entre cette représentation et les informations issues des capteurs. *"A sensor-tuned-representation is a data structure designed for the specific purpose of facilitating the recognition of objects and the determination of their position and orientation from data provided by a given sensor or a set of sensors. By definition, therefore, a sensor-tuned-representation for, say, a structured light 3D vision sensor is different from that for a 2D photometric sensor, or a tactile sensor. Essentially, a sensor-tuned-representation is an intermediate representation that can be derived from both high-level and low-level representations and facilitates the matching of partial object models with complete object models."*

##### 4.1.5.1. Représentation spéciale capteur (sensor-tuned-representation)

Kak utilise un capteur 3D fonctionnant avec une lumière structurée laser. Ce type de senseur génère des informations surfaciques incluant contour de face, position, orientation, relation entre surfaces et bord séparant deux surfaces. Les modèles d'objets sont donc spécialement décrits avec ce type de primitives visuelles et ont la forme de graphes relationnels à attributs.

Exemple de structure de modèle :

- object (name,type,surfaces,relations)
- surface (name,type,area,position,orientation,attributes)
- relation (name,type,attributes,surface 1,surface2)

On constate dans cette modélisation l'absence totale de critères visuels 2D. Ceci est du à l'utilisation d'un capteur 3D capable de fournir des informations directement exploitables en 3D. Il n'est évidemment pas nécessaire d'utiliser un "générateur d'aspect 2D".

#### 4.1.5.2. Segmentation d'image 3D et comparaison

Nous avons vu précédemment que l'élaboration des modèles d'objet est simplifiée par une définition dans un espace à trois dimensions. Cette simplification à ce niveau haut provoque en contre partie une augmentation de la complexité du traitement d'image au niveau bas puisqu'il faut que ce dernier soit capable de fournir une segmentation 3D de l'image. Ceci entraîne donc une complication des traitements "en ligne" au profit des traitements "hors ligne". Le système de Kak décrit l'image sous forme d'un graphe et applique un algorithme spécifique de comparaison de graphes pour identifier l'objet présent sur la scène.

#### 4.1.6. Conclusion sur les systèmes de conception des "connaissances" consommées par un système de vision et proposition d'un MCD

La compréhension d'image est une succession de structurations d'informations, partant, au niveau bas, d'une information à l'état brute et parvenant, au niveau haut, à une information moins nombreuse et plus complexe.

La synthèse d'image [PUE 87] est le processus inverse de la compréhension d'image, qui consiste à déstructurer une information de haut niveau (une définition 3D volumique d'un objet) en un certain nombre de niveaux plus bas (projection selon différents plans).

Comme le dit [KAK 87], en pratique, le traitement d'image fait intervenir en amont un processus de synthèse d'image (modélisation) et en aval un processus de compréhension d'image (segmentation, interprétation). Ces deux processus opposés se rejoignent en un niveau de représentation intermédiaire privilégié et dépendant des senseurs utilisés. Les algorithmes de comparaison s'appliquent à ce niveau particulier de représentation.

La phase de "prédiction" de ce que le capteur doit voir, aboutit à la création de ce que l'on appelle un "modèle d'objet" (*individu* MODELE VISUEL), fruit de la synthèse.

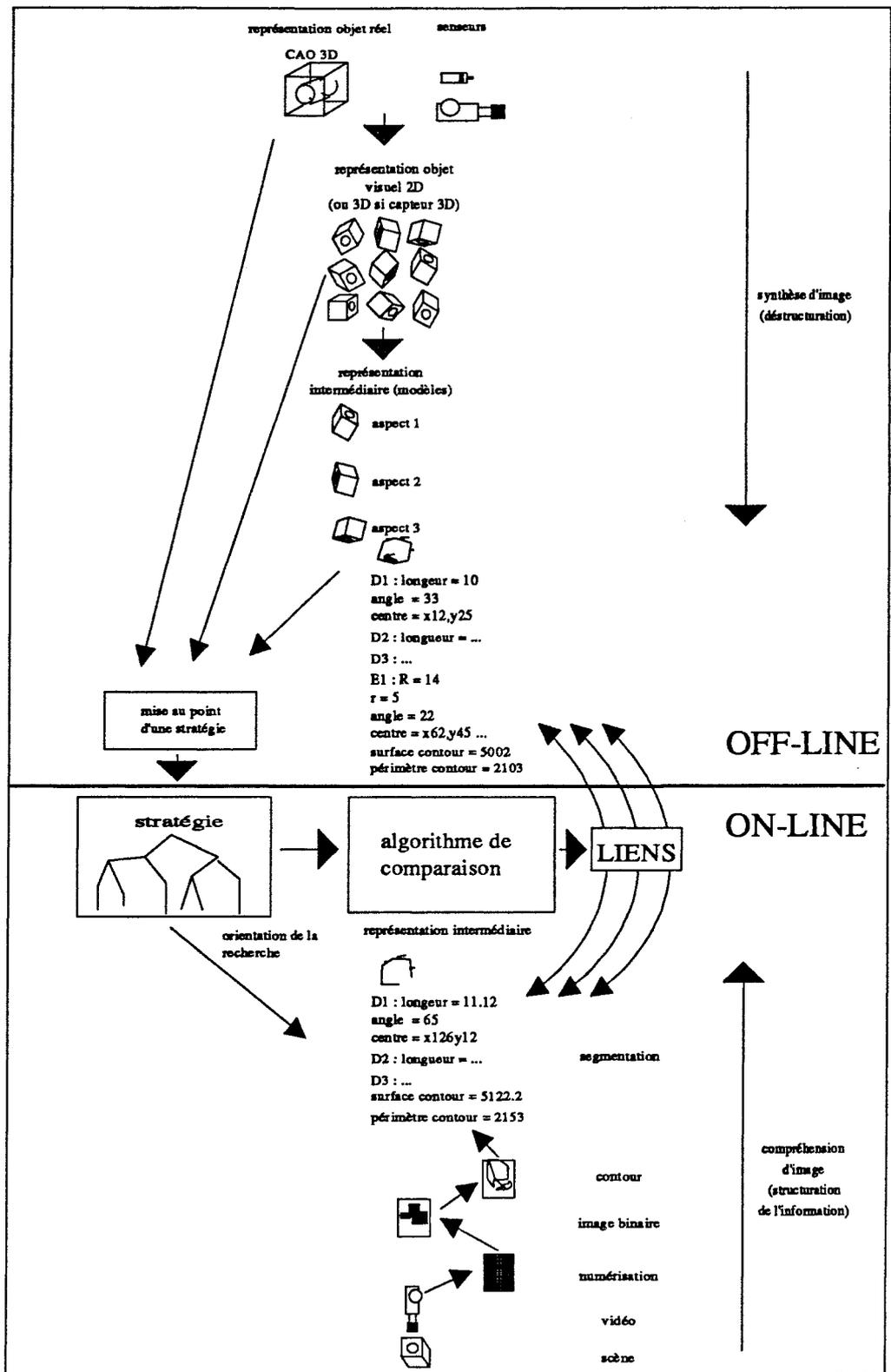


figure 4.4. Fonctionnement d'un système de vision

Cette "prédiction" se fait de diverses manières : par passage de l'objet sous la caméra (ou sous le capteur utilisé) afin de recueillir un signal réel qui sert de modèle [CIC 87], par calculs théoriques réalisés par le concepteur du système (calculs de surface, de périmètre etc...) à partir de plans de l'objet, ou dans le cas de systèmes plus intégrés, les modèles peuvent être générés automatiquement (mais presque toujours partiellement) à partir de modèles CAO de l'objet.

Même dans les traitements très simples tel que comptage de pixels ou comparaison d'image (pattern matching), on est obligé de passer par cette phase de prédiction (prédire le nombre de pixels de référence, trouver une image modèle idéale).

Pour réaliser cette synthèse d'image (ou de modèle), le concepteur doit pouvoir disposer des informations suivantes (cf. figure 4.5.) :

- définition 3D de la forme de l'objet, de son environnement et de sa position par rapport au capteur, (*individu* MONTAGE SCENE),
- description des éléments d'acquisition du signal du système de vision employé, (*individus* MONTAGE OPTIQUE, MONTAGE ECLAIRAGE, MONTAGE CAPTEUR).

Les *relations* entre ces *individus* et les *individus* MODELE VISUEL, MODELE SEGMENTATION et MODELE PRETRAITEMENT contribuent à décrire la sémantique du MCD de Référence du système de vision.

L'ensemble des traitements qui aboutissent à la conception des algorithmes de vision appartiennent à la classe des systèmes génératifs.

Ils nécessitent énormément de sémantique.

Celle-ci est implicitement intégrée dans les traitements, c'est pourquoi notre proposition de MCD peut paraître "légère" en regard de l'énormité de la tâche effectuée.

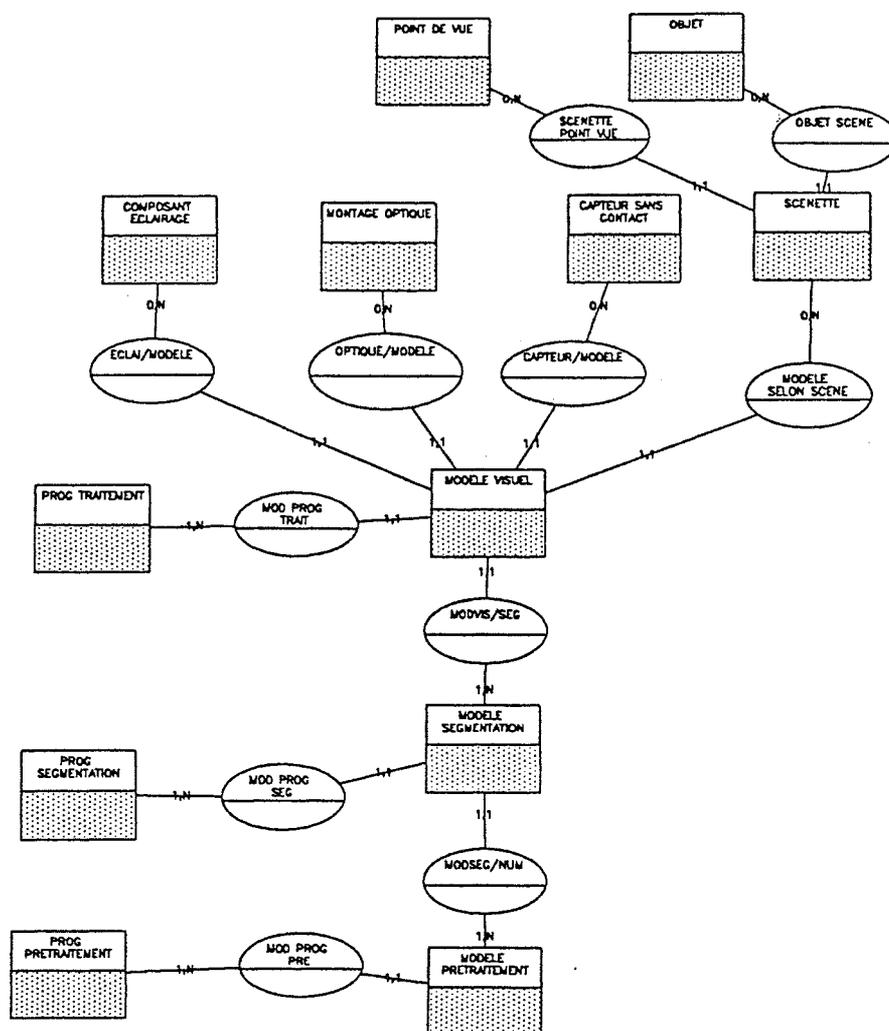


figure 4.5. Extrait du MCD de vision : les données nécessaires à la conception des connaissances consommées par un système de vision

## 4.2. Un cycle de conception d'un processus de contrôle par vision

### 4.2.1. Le besoin en génie visionique

"La situation de l'ingénierie visionique est assez semblable à celle du génie automatique".

[VOG 90] montre dans quelle mesure il est intéressant d'utiliser les outils du Génie Automatique comme support (ou exemple) dans le Génie Visionique. Il est bien évident que ces outils et méthodes ne peuvent apporter aucun "plus" à l'ingénierie de vision en tant que système de traitement d'image, mais plutôt en tant que systèmes automatisés intégrables.

*"Nous assimilons le système de vision à un automatisme à part entière qui possède son propre cycle de vie".*

#### **4.2.1.1. La notion de cycle de vie**

En génie logiciel, le cycle de vie se définit comme *"une modélisation conventionnelle de la succession d'étapes qui préside à la mise en oeuvre d'un produit logiciel"*.

Trois interprétations principales de la notion de cycle de vie sont possibles :

- *"interprétation neutre qui considère le CDV comme une description idéalisée de ce qui se passe dans un projet logiciel, et qui se veut être un élément de référence, sans plus*
- *interprétation volontariste qui veut faire respecter le modèle dans l'organisation du projet, en imposant des outils-méthodes à utiliser (SADT, MERISE, ...) à chaque étape et les relations obligatoires entre ces étapes*
- *interprétation intermédiaire qui consiste à définir, pour chaque projet, un CDV de référence, le mieux adapté au sujet traité ou au type d'application. On parle alors de CDV "adaptatif" ou "évolutif". L'intérêt de cette dernière interprétation est de formaliser la démarche entreprise sur un projet particulier, afin de la mémoriser pour pouvoir la réutiliser et éventuellement la faire évoluer avec l'expérience accumulée sur un type de projet; c'est en quelque sorte une manipulation du savoir faire" [MEY 86].*

*"Méthodologie : ensemble structuré et cohérent de méthodes, guides, outils permettant de déduire la manière de résoudre un problème.*

*Méthode : technique de résolution de problème, caractérisée par un ensemble de règles bien définies qui conduisent à une solution correcte" [CAL 89].*

#### **4.2.1.2. La réutilisabilité**

Cette notion est directement issue du Génie logiciel. La réutilisabilité consiste en la réalisation de petits composants de comportement bien maîtrisés à forte cohérence interne et faible couplage externe présentant trois avantages principaux :

- réduction des coûts de développement
- amélioration de la qualité du logiciel
- accélération de la production du logiciel.

#### **4.2.1.3. Les objets**

Un objet est une entité qui a un état et dont le comportement est défini par les actions qu'il peut subir ou les services qu'il peut demander à d'autres objets. L'objet est l'application parfaite de la notion de réutilisabilité en automatique. La notion d'objet rejoint les travaux antérieurs de Modélisation de Partie Opérative et de Structuration de la Partie Commande selon le concept de Filtre de Comportement [LHO 85] et trouve actuellement son prolongement dans le concept d'objets "intelligents" [IUN 91].

#### **4.2.2. Application à la visionique**

[VOG 90] propose l'utilisation d'un cycle de vie en V.

A chaque étape du cycle de vie de conception de l'application vision est associée une méthode et son outil (selon l'idée du Poste de Travail pour l'Automaticien - PTA).

*"Les modèles formalisent le savoir-faire, condensant en quelque sorte l'expérience acquise lors de réalisations antérieures; ils peuvent être amenés à évoluer au cours du temps et sont en principe dédiés à une classe de problèmes. L'instanciation de ces modèles à des cas particuliers facilite la résolution des problèmes et homogénéise les solutions"* [VOG 90].

Cette méthodologie est située sur les axes Dérivation et Généralisation/Particularisation. L'axe Génératif n'est pas du tout exploité (sauf dans les phases de conception des traitements de vision).

Elle comporte quatre phases : spécification, analyse, conception et codage. Les deux premières phases constituent un tronc commun aux différents "métiers" (optique, mécanique, automatique, vision, algorithmique) que nécessite la mise en place d'un système de vision alors que la phase de conception et de codage est spécifique à chaque "métier".

#### **4.2.2.1. La phase de spécification**

La phase de spécification des besoins à pour but tout d'abord d'élaborer un cahier des charges fonctionnel du système de vision. Pour ce faire, Vogrig préconise l'utilisation d'un modèle de cahier des charges proposé par le "club vision français". La formalisation des besoins qui suit utilise la méthode SADT (associée avec ORCHIS, l'outil informatique qui supporte SADT).

[VOG 90] distingue la formalisation des besoins d'un point de vue service (préparation, exploitation et maintenance du capteur) et d'un point de vue procédé (objet de l'implémentation du capteur : visionique pure).

#### **4.2.2.2. La phase d'analyse**

La phase d'analyse comprend :

- L'analyse fonctionnelle dont le but est de formaliser le comportement, c'est à dire la façon dont le système de vision devrait marcher. La formalisation du comportement permet d'exploiter le concept de Génie Logiciel et Génie Automatique de **réutilisabilité**. La notion d'actinomie est utilisée pour formaliser le comportement du système de vision (modèle dynamique), SADT pour l'analyse de la structure fonctionnelle de l'automatisme (modèle statique),
- L'étude de faisabilité utilise des modèles d'applications existantes supportés par des plates-formes de développement spécifique à la vision (visilog de noésis, système expert d'aide à la conception d'application de vision [III 89]). [VOG 90] rappelle que l'étude de faisabilité est indispensable. La réalisation d'un

prototype mis en situation réelle doit être systématique tant les difficultés sont nombreuses lors de la mise au point d'un système vision (du fait de sa sensibilité à l'environnement).

#### **4.2.2.3. La phase de conception**

La phase de conception concerne les différents "métiers" de la visionique. Les méthodes de conception des "connaissances" consommées par les systèmes de vision que nous avons exposées précédemment conviennent parfaitement pour la conception des traitements de vision, de même que des méthodes existent également pour la conception des optiques. [VOG 90] n'a traité que la conception de la Partie Commande. Il exploite la méthode et les outils du Génie Automatique que nous avons utilisés au chapitre II : la méthode de structuration de la partie commande [LHO 87] associée à l'outil ORCHIS-GRILLE.

#### **4.2.2.4. La phase de codage**

Les méthodes de codage varient selon les matériels utilisés. Une CAO Grafcet CADEPA est utilisée pour la génération du code automate. Les programmes de vision sont développés en exploitant au maximum les capacités d'environnement de développement fournis par les systèmes d'exploitation des systèmes informatiques utilisés.

### **5. Notre approche pour le prototypage du contrôle-commande de niveau station intégrant le procédé de vision**

*"L'objectif est de permettre une programmation intégrée de la cellule à partir des données du produit" [MUN 88][ORT 88].* Nous voulons, en partant du niveau le plus haut (définition du produit), être capable de prototyper le contrôle-commande de la station pour l'exécution d'une tâche de vision dans le cadre d'une cellule flexible en suivant une méthodologie associée à des outils nous permettant de réutiliser notre savoir faire.

Dans le chapitre III nous avons utilisé l'outil ORCHIS-GRILLE pour concevoir la partie commande des équipements de vision (capteur, éclairage); dans ce paragraphe nous proposons une méthodologie de conception de la partie commande de la station appliquée à une tâche de vision.

Principe de la méthodologie :

- étape 1 : générer la gamme d'usinage avec l'outil PROPEL (approche générative)
- étape 2 : intégrer les opérations de contrôle dans la gamme (approche dérivative)
- étape 3 : détailler et valider la gamme sur l'outil SPEX (approche instanciative)

Nous détaillons cette méthodologie à travers un exemple : réaliser le contrôle des quatre perçages de la pièce NASA sur le CHARLY-ROBOT.

## **5.1. Prototypage du contrôle par vision des perçages de la pièce NASA sur le CHARLY-ROBOT**

### **5.1.1. Génération de la gamme d'usinage par PROPEL**

La pièce NASA est décrite en "entités du gammiste" (figure 4.6.).

La gamme d'usinage proposée par le système expert PROPEL (cf. figure 4.7.) pour un atelier composé d'un tour et d'une fraiseuse se compose de 2 phases. La phase 10 effectue l'usinage de la face "FACEDESSOUS" qui sert d'appui à la phase 20 qui réalise l'ensemble des opérations d'usinage en une seule sous-phase (sans démontage de la pièce). Nous ne nous intéressons qu'à une partie de cette sous-phase, celle qui concerne les ébauches et finition des entités TROULAME, TROU3, TROU2 et TROU1.

```

;Description generale de la piece du 18.09.91
(part NASA (ra 1.6) (quality 10) (s-x 60) (s-y 60) (s-z 40))
; Description des entites
(feature Facedessous
  (type face)
  (normal z-)
  (ra 0.8)
  (s-x 60)
  (s-y 60))
(feature Febauche
  (type cluttered-face)
  (normal z+)
  (ra 0.8)
  (s-x 60)
  (s-y 60)
  (support-of Facel Contour1))
(feature Facel
  (type boss-face)
  (normal z+)
  (compose bossel)
  (ra 0.8)
  (s-x 50)
  (s-y 50)
  (support-of Fcercle Ccercle))
(feature Fcercle
  (type boss-face)
  (normal z+)
  (compose bosse2)
  (s-x 50)
  (s-y 50)
  (support-of Ftriangle Ctriangle))
(feature Ftriangle
  (type boss-face)
  (normal z+)
  (compose bosse3)
  (s-x 44)
  (s-y 38))
(feature Contour1
  (type contour)
  (compose bossel)
  (normal z+))
(feature Ccercle
  (type contour)
  (compose bosse2)
  (normal z+))
(feature Ctriangle
  (type contour)
  (compose bosse3)
  (normal z+))
(feature bossel
  (type boss)
  (support Febauche)
  (face Facel)
  (lateral-surface Contour1))
(feature bosse2
  (type circular-boss)
  (support Facel)
  (face Fcercle)
  (lateral-surface Ccercle))
(feature bosse3
  (type boss)
  (support Fcercle)
  (face Ftriangle)
  (lateral-surface Ctriangle))
(feature trou1
  (type bore)
  (normal z)
  (diameter 6)
  (depth 15))
(feature trou2
  (type bore)
  (normal z)
  (diameter 6)
  (depth 15))
(feature trou3
  (type bore)
  (normal z)
  (diameter 6)
  (depth 15))
(feature trouleme
  (type counterbore-hole)
  (normal z)
  (diameter 6)
  (depth 15)
  (counterbore-diameter 14)
  (counterbore-depth 5))

```

*figure 4.6. Définition par entité du gammiste de la pièce NASA*

10 PHASE	LATHE
SOUS-PHASE	FEBAUICHE
OPERATION	FACING-TOOL
EBAUCHE	FACEDESSOUS
OPERATION	FACING-TOOL
FINITION	FACEDESSOUS
20 PHASE	MC-TRAD
SOUS-PHASE	FACEDESSOUS
OPERATION	SURFACING-CUTTER
EBAUCHE	FTRIANGLE
EBAUCHE	FCERCLE
EBAUCHE	FACE1
OPERATION	SURFACING-CUTTER
FINITION	FTRIANGLE
FINITION	FCERCLE
FINITION	FACE1
OPERATION	DRILL
EBAUCHE	TROULAME
OPERATION	DRILL
EBAUCHE	TROU3
EBAUCHE	TROU2
EBAUCHE	TROU1
OPERATION	REAMING-TOOL
FINITION	TROU3
FINITION	TROU2
FINITION	TROU1
OPERATION	FACE-CUTTER
EBAUCHE	CTRIANGLE
OPERATION	FACE-CUTTER
FINITION	CTRIANGLE
OPERATION	FACE-CUTTER
EBAUCHE	CCERCLE
OPERATION	FACE-CUTTER
FINITION	CCERCLE
OPERATION	FACE-CUTTER
EBAUCHE	CONTOUR1
OPERATION	FACE-CUTTER
FINITION	CONTOUR1
OPERATION	FACE-CUTTER
EBAUCHE	FEBAUICHE
OPERATION	FACE-CUTTER
FINITION	FEBAUICHE
OPERATION	REAMING-TOOL
EBAUCHE	TROULAME
OPERATION	COUNTERBORING-CUTTER
FINITION	TROULAME

*figure 4.7. Gamme d'usinage de la pièce NASA*

### 5.1.2. Insertion des opérations de contrôle dans la sous-phase

La partie de la sous-phase d'usinage que nous voulons traiter est :

OPERATION	DRILL
EBAUCHE	TROULAME
OPERATION	DRILL
EBAUCHE	TROU3
EBAUCHE	TROU2

EBAUCHE	TROU1
OPERATION	REAMING-TOOL
FINITION	TROU3
FINITION	TROU2
FINITION	TROU1

L'insertion des opérations de contrôle nous oblige à casser cette gamme d'usinage, en effet le système expert PROPEL propose de réaliser les trois perçages en une seule opération (sans changement d'outil); mais comme il est nécessaire de contrôler chaque perçage indépendamment et immédiatement après son exécution (en cas de bris d'outil dans le perçage de TROU3, il serait dangereux de faire TROU2 et TROU1 sans effectuer de changement d'outil).

*La sous-phase d'usinage devient une sous-phase d'usinage et de contrôle :*

OPERATION	DRILL
EBAUCHE	TROULAME
OPERATION	CAMERA
CONTROLE	TROULAME
OPERATION	DRILL
EBAUCHE	TROU3
OPERATION	CAMERA
CONTROLE	TROU3
OPERATION	DRILL
EBAUCHE	TROU2
OPERATION	CAMERA
CONTROLE	TROU2
OPERATION	DRILL
EBAUCHE	TROU1
OPERATION	CAMERA
CONTROLE	TROU1
OPERATION	REAMING-TOOL
FINITION	TROU3
OPERATION	CAMERA
CONTROLE	TROU3

OPERATION	REAMING-TOOL
FINITION	TROU2
OPERATION	CAMERA
CONTROLE	TROU2
OPERATION	REAMING-TOOL
FINITION	TROU1
OPERATION	CAMERA
CONTROLE	TROU1

Cette *sous-phase d'usinage et de contrôle* sert de base à la Programmation Fonctionnelle.

### 5.1.3. Prototypage sur SPEX

#### 5.1.3.1. Programmation Fonctionnelle

Programmer fonctionnellement la fabrication [PAR 90] revient à décrire toutes les séquences d'actinomies élémentaires correspondant aux opérations élémentaires de la gamme de fabrication (incluant l'usinage, le contrôle et la manutention). Une actinomie comprend trois actèmes : la préparation, le noyau et la cloture [VOG 89].

Une actinomie exprime une séquence d'activités permettant d'atteindre un objectif déterminé. La formalisation d'une actinomie suppose une bonne connaissance du processus que l'on représente. L'ordonnancement figé des activités et leur caractère antinomique assurent la prise en compte non seulement des activités de préparation, mais également des activités de clôture :

- préparation
  - réalisation de l'objectif
- clôture

Les actinomies peuvent fusionner entre elles. La fusion des actinomies suppose l'activité "fusionnante" d'un niveau haut et globalisant, assurant une conduite cohérente des niveaux élémentaires. Réaliser cette fusion revient à

réaliser la commande d'un ensemble d'équipements dont les comportements élémentaires locaux sont connus mais dont le comportement global (ensembliste) reste à définir. La fusion en série ne pose pas de problème (activité globale séquentielle), par contre la fusion en parallèle (activité globale quelconque) demande la mise au point d'algorithmes évolués faisant appel à une analyse sémantique de la gamme d'usinage.

Le système de vision est un ensemble d'équipements, ayant des comportements élémentaires définis, intervenant dans un contexte large de fabrication dont le comportement global est la fusion (au sens large) des comportements des procédés de fabrication, à savoir : vision, usinage, transport, palpéage etc ... Dans l'architecture de la cellule telle qu'elle a été prévue dans le projet, cette fusion s'effectue au niveau station. On peut décomposer le système de vision en plusieurs sous-systèmes : l'éclairage, le porte-captéur, le porte-objet (NUM), le captéur (carte de numérisation + caméra), le noyau d'analyse d'image.

Nous proposons une actinomie du système de vision représentant son comportement appliqué à la classe d'application des "contrôles sur le CHARLY-ROBOT". Elle exprime les contraintes résultant du fonctionnement global des divers équipements intervenant dans l'opération de vision. Par exemple, l'analyse de l'image ne peut se faire sans avoir capté l'image, l'image ne peut être captée sans avoir allumé l'éclairage et amené la caméra et l'objet.

#### Actinomie de contrôle par vision sur le CHARLY-ROBOT :

- amener captéur en position de prise de vue (NUM)
  - allumer éclairage
    - captéur image (prétraitements compris)
  - éteindre éclairage
- amener captéur en position de repli
  
- analyser image

### 5.1.3.2. Fusion des actinomies

Nous présentons l'actinomie d'un perçage sur le CHARLY-ROBOT. Cette fraiseuse didactique ne dispose pas de système automatisé de changement d'outil; le montage et le démontage d'outil se fait manuellement.

- monter outil (manuellement)
  - déplacer chariot porte broche au rep usinage
    - approcher outil -> rep approche
      - exécuter macro d'usinage
    - dégager outil -> au rep degagement
  - déplacer chariot porte broche au rep changement d'outil
- démonter outil (manuellement)

Nous présentons un exemple de Programmation Fonctionnelle sur une partie de la sous-phase de contrôle établie précédemment.

La partie de gamme de contrôle est :

OPERATION	DRILL
EBAUCHE	TROU3
OPERATION	CAMERA
CONTROLE	TROU3
OPERATION	DRILL
EBAUCHE	TROU2
OPERATION	CAMERA
CONTROLE	TROU2
OPERATION	DRILL
EBAUCHE	TROU1
OPERATION	CAMERA
CONTROLE	TROU1

La Programmation Fonctionnelle de cette sous-phase est :

- monter outil (manuellement)
  - déplacer chariot porte broche au rep usinage
    - approcher outil -> rep approche
      - ébauche perçage TROU3
    - dégager outil -> au rep degagement
  - déplacer chariot porte broche au rep changement d'outil
- démonter outil (manuellement)
  - amener capteur en position de prise de vue (NUM) TROU3
    - allumer éclairage
      - capter image (prétraitements compris)
    - éteindre éclairage
  - amener capteur en position de repli
  - analyser image
- monter outil (manuellement)
  - déplacer chariot porte broche au rep usinage
    - approcher outil -> rep approche
      - ébauche perçage TROU2
    - dégager outil -> au rep degagement
  - déplacer chariot porte broche au rep changement d'outil
- démonter outil (manuellement)
  - amener capteur en position de prise de vue (NUM) TROU2
    - allumer éclairage
      - capter image (prétraitements compris)
    - éteindre éclairage
  - amener capteur en position de repli
  - analyser image
- monter outil (manuellement)
  - déplacer chariot porte broche au rep usinage
    - approcher outil -> rep approche
      - ébauche perçage TROU1
    - dégager outil -> au rep degagement
  - déplacer chariot porte broche au rep changement d'outil
- démonter outil (manuellement)

- amener capteur en position de prise de vue (NUM) TROU1
  - allumer éclairage
    - capter image (prétraitements compris)
  - éteindre éclairage
- amener capteur en position de repli
- analyser image

Cette actinomie peut être réalisée sans aucune intervention cognitive importante, il suffit de mettre "bout à bout" les actinomies des procédés concernés en appliquant la relation d'ordre donnée par la sous-phase. La fusion de cette actinomie (l'optimisation) exige au contraire une activité intellectuelle plus importante, nécessitant une connaissance globale des différents procédés engagés. On obtient après fusion (manuelle) en série :

- monter outil (manuellement)
  - déplacer chariot porte broche au rep usinage
    - approcher outil -> rep approche
      - ébauche perçage TROU3
  - dégager outil -> au rep degagement
  - amener capteur en position de prise de vue (NUM) TROU3
    - allumer éclairage
      - capter image (prétraitements compris)
    - éteindre éclairage
  - analyser image
  - approcher outil -> rep approche
    - ébauche perçage TROU2
  - dégager outil -> au rep degagement
  - amener capteur en position de prise de vue (NUM) TROU2
    - allumer éclairage
      - capter image (prétraitements compris)
    - éteindre éclairage
  - analyser image
  - approcher outil -> rep approche
    - ébauche perçage TROU1
  - dégager outil -> au rep degagement

- amener capteur en position de prise de vue (NUM) TROU1
  - allumer éclairage
    - capter image (prétraitements compris)
  - éteindre éclairage
- analyser image
- amener capteur en position de repli

L'actinomie fusionnée représente le comportement idéal, sans dysfonctionnement, à l'image du graphe principale (F1) dans la définition des modes de Marche et d'Arrêt au sens GEMMA [GEM AD]. En cas de problème, cette actinomie fusionnée devient inopérante car elle n'inclut pas les modes dégradés et il faut reprendre le comportement des actinomies non fusionnées. Ce raisonnement constitue une contribution de [MOR 92] au GEMMA. Nous utilisons l'outil SPEX pour prototyper à la fois le comportement fusionné de l'ensemble des procédés mais aussi pour prototyper les modes dégradés.

### **5.1.3.3. Utilisation sur SPEX**

#### **5.1.3.3.1. L'application**

L'application comprend deux niveaux : station et équipement (cf. figure 4.8.).

##### **5.1.3.3.1.1. Le niveau équipement (environnement)**

Nous réutilisons les Boîtes Fonctionnelles instanciées que nous avons développées lors de la conception des équipements (cf. chapIII). Ces modèles d'équipements servent d'environnement à la station dans notre application. Nous simulons : deux lampes, le capteur (edge 90), le traitement d'image, l'usinage et le palpé. Ces équipements sont réimplantés dans l'application sous forme de Diagrammes Fonctionnels. Ils sont tous connectés au Diagramme Fonctionnel de la station.

#### **5.1.3.3.1.2. le niveau station (système)**

La partie commande de la station est implantée dans un Diagramme Fonctionnel. L'interface d'entrées/sorties de ce Diagramme Fonctionnel a été définie au chapitre III avec l'outil ORCHIS-GRILLE. Notre objectif est de modéliser son comportement à partir de la gamme de contrôle.

#### **5.1.3.3.2. Prototypage du contrôle-commande de la station**

##### **5.1.3.3.2.1. Démarche**

Notre point d'entrée pour prototyper la programmation de la station est la sous-phase de contrôle. Elle représente le déroulement normal des opérations d'usinage et de contrôle, malheureusement, elle ne nous renseigne aucunement sur les modes dégradés à exécuter en cas de bris d'outil ou d'impossibilité de réaliser un contrôle. Nous distinguons deux niveaux fonctionnels :

- *le niveau opération (procédé)* : ce niveau est représenté par un ensemble de Boîtes Fonctionnelles chargées de contrôler une opération complète; elles incluent donc leur propre actinomie de fonctionnement normal, mais aussi leur mode de marche dégradée. Ainsi, pour la Boîte Fonctionnelle générique "contrôler un perçage par vision", si une lampe vient à tomber en panne, le système de contrôle commande de l'opération essaie de résoudre le problème localement. Si l'opération de contrôle est irréalisable, le défaut doit engendrer une réaction au niveau fonctionnel supérieur : le niveau sous-phase,
- *le niveau sous-phase (station)* : ce niveau est modélisé par une Boîte Fonctionnelle particulière à la sous-phase (il n'y a a priori pas de sous-phase générique). Le comportement normal de cette boîte est directement donné par la définition de la sous-phase de contrôle. L'outil SPEX nous permet de prototyper les différents modes de marche de la réalisation d'une sous-phase (relancer une opération, effectuer un changement d'outil, annuler la sous phase, etc ...).



Nous voulons concevoir la Partie Commande de la station par assemblage de Boîtes Fonctionnelles modélisant les comportements de différents procédés (contrôle par vision, palpé, usinage, manutention). Ces modules possèdent leur comportement propre, il nous suffit de les paramétrer et de les connecter à la manière de circuits intégrés pour prototyper la commande de la station.

#### **5.1.3.3.2.2. Boîte Fonctionnelle "CONTROLE PAR VISION"**

Cette Boîte Fonctionnelle modélise le comportement global d'un système de vision (cf. figure 4.9.), elle a une structure de type "surveiller, réagir, commander".

Le module "surveiller" permet d'assurer la **validation technologique** du capteur par vision (un contrôle ne sera validé que si toutes les étapes de sa réalisation se sont passées correctement). Le "bon comportement" (modélisé par l'actinomie de vision présentée plus haut) est programmé dans le grafcet "commander", le grafcet "réagir" permet d'intégrer les modes de marche tels que stopper le contrôle si la caméra est hors-service, allumer la lampe n°2 si la lampe n°1 est hors d'usage, dégager la caméra si il y a un problème de prise d'image, etc ...

#### **5.1.3.3.2.3. Boîte Fonctionnelle "USINAGE"**

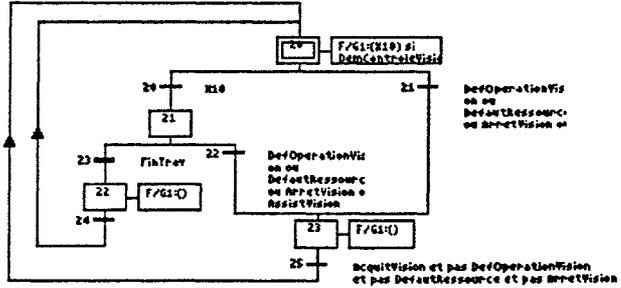
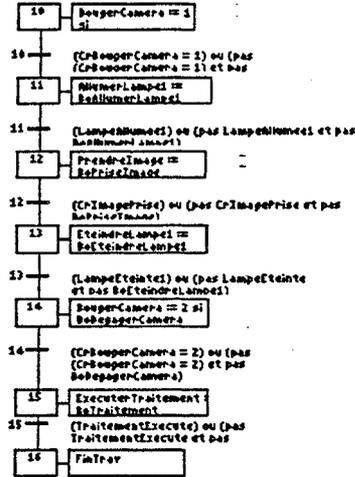
Cette Boîte Fonctionnelle modélise le comportement du procédé d'usinage (cf. figure 4.10.).

#### **5.1.3.3.2.4. Boîte Fonctionnelle des autres procédés**

Les procédés de contrôle par palpé RENISHAW, de contrôle par vision linéaire, de montage et démontage de la pièce, etc ... font également l'objet d'une modélisation de leur mode opératoire. L'objectif est de disposer d'une **bibliothèque de comportement opératoire permettant** de concevoir la commande globale des ces procédés par la station.

Grafset : Procède-Vision

Fichiers	Creation	Outils	Simulation	Bibliothèques
----------	----------	--------	------------	---------------



	↑ ↓	entrees	↑ ↓	C.E.
	LampeAllumee1 (Bool) LampeEteinte1 (Bool) DefTrayNum (Bool) CrImagePrise (Bool) DefImage (Bool) TraitementExecute (Bool)		DefOperationVision AllumerLampe1 EteindreLampe1 PrendreImage ExecuterTraitement BougerCamera	
	DefOperationVision si (DefautLampe1 ou DefTrayNum ou DefImage ou DefTraitement)			Commente

figure 4.9. Commande du procédé de vision sur le CHARLY-ROBOT

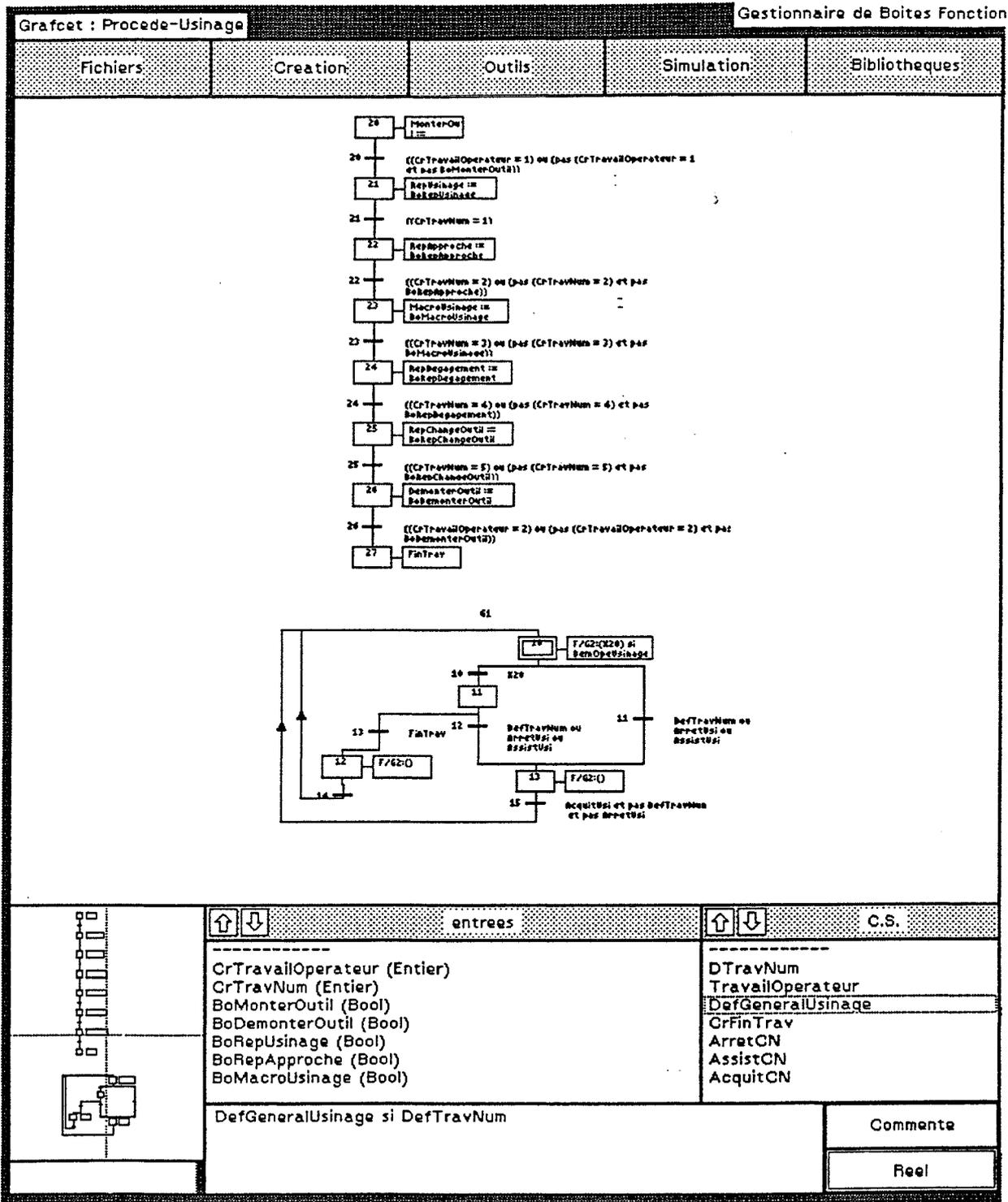


figure 4.10. Commande du procédé d'usinage sur le CHARLY-ROBOT

#### 5.1.3.3.2.5. Fusion des Boîtes Fonctionnelles

La fusion des comportements des Boîtes Fonctionnelles est réalisée par un paramétrage du grafset "commander" de chacune des boîtes. A chaque activité correspond une variable que l'on peut fixer de manière externe et qui spécifie si l'activité doit être réalisée ou non. L'ensemble de ces variables constitue un "bornier" qui permet de contraindre le fonctionnement du procédé. Le "bornier" de la Boîte Fonctionnelle "contrôler par vision" comporte 6 pines : "positionner caméra, allumer lampe, prendre image, éteindre lampe, dégager caméra, traiter l'image". La configuration du "bornier" ne peut en aucun cas modifier les modes de marche du procédé concerné.

L'utilisation du "bornier" permet de prototyper la **commande optimum du processus opératoire dans la phase de bon fonctionnement** en intégrant la fusion des activités antinomiques sans pénaliser la génération des différents modes de marche largement tributaires du savoir-faire de chaque procédé (vision, usinage ...) et encapsulés dans la Boîte Fonctionnelle "procédé".

#### 5.1.3.3.2.6. Boîte Fonctionnelle "SOUS-PHASE"

Cette Boîte Fonctionnelle modélise le comportement global de la station lors de l'exécution de la sous-phase (cf. figure 4.11.). C'est donc elle qui commande aux Boîtes Fonctionnelles "contrôler par vision" et "usiner". Elle gère la fusion des comportements en paramétrant les "borniers" de chaque procédé. Elle a elle même une structure du type "surveiller réagir commander"; elle peut donc gérer des mode dégradés de sous-phase. Ces modes ne sont pas prévus par la gamme que génère PROPEL, ils sont donc conçus à partir du savoir-faire du prototypeur et viennent compléter la gamme initiale.



- la gamme de contrôle (générée manuellement)

de valider la commande de la station qui englobe :

- les modes de marche du procédé de vision
- les modes de marche du procédé d'usinage
- les modes de marche de l'exécution d'une sous-phase par la station

en réutilisant les modèles de commande générés lors de la conception des équipements (éclairage, capteur, NUM) et en utilisant une bibliothèque de comportements de niveau procédé (Boîte Fonctionnelle générique "contrôle par vision", "usinage"). Ce stockage du savoir faire contribue à simplifier et à flexibiliser la conception de la commande de la station.

L'ensemble des comportements prototypés sous SPEX font ensuite l'objet d'un codage en langage C sous OS2 dans la station. Chaque Boîte Fonctionnelle (procédé vision, procédé usinage...) fait l'objet de la programmation d'une tâche OS2 [DOI 92] au sein du programme de pilotage station (dans les modules TQM).

L'outil SPEX devient une extension du générateur de gamme PROPEL, ces travaux vont dans le sens du projet PRIMECA.

## Conclusion

Le Système d'Informations peut se définir de la manière suivante : " *le SI d'une organisation sociale est l'ensemble des moyens, humains et matériels, et des méthodes se rapportant au traitement des différentes formes d'information rencontrées dans les organisations*" [GAL 86].

"*Le rôle primordial du SI, outre le stockage et la communication, est la transformation de données en informations*" [COU 91]. Les informations reflètent l'état du Système Opérant. Le Système Décisionnel exploite les informations "d'état". Le Système d'Information constitue donc un tampon entre Système Décisionnel et Système Opérant; il est en quelque sorte "l'organe sensoriel" du Système Décisionnel. Les programmes de traitement par vision qui interprètent les "données" d'une image binaire pour en extraire des informations pertinentes pour le pilotage de la cellule font partie intégrante du Système d'Information, organe sensoriel de la cellule.

Nous avons déterminé pour les *domaines* de la conception, de la gestion technique et de la fabrication l'ensemble des Modèles Conceptuels de Données de Référence exprimés en formalisme Entité/Relation étendu appliqués à la technologie vision. Nous avons montré à travers l'application concrète de la cellule MRT comment ces modèles peuvent être particularisés en MCD particuliers, puis dérivés en MOD, MLD et Modèle Physique de données, participant ainsi à la réalisation du Système d'Information du Système Intégré de Production.

L'ensemble des MCD de Référence que nous avons conçus constitue un premier pas vers une démarche structurée d'intégration de la vision. La généralité obligatoire de ces modèles de référence (ils doivent s'appliquer à tous les systèmes de vision) ne nous a pas autorisé à détailler tous les types de matériels existant sur le marché mais nous a imposé d'en extraire l'invariant conceptuel propre à tous système de vision. La modélisation de la sémantique des systèmes

de vision ne peut se faire de manière "rentable" que dans le cadre d'un modèle particulier. La réalisation de ces MCD de Référence sert de réservoir de savoir faire en matière de conception du Système d'Information intégrant la vision. Elle ne préjuge en aucun cas de l'utilisation de tel ou tel matériel ni d'une quelconque répartition sur des sites; ces considérations n'intervenant que dans les étapes ultérieures de la conception d'un Système d'Information Particulier.

Nous avons également proposé une architecture physique d'intégration de la vision dans la cellule flexible respectant les qualité de flexibilité, d'intégrabilité et d'auto-contrôle du système de vision ainsi qu'une démarche de conception de l'ensemble de son automatisation, allant du prototypage des sous-phases jusqu'à la spécification des parties commandes de chaque équipement de vision. Ces propositions ont fait l'objet de réalisations concrètes. L'utilisation de méthodes issues du Génie Automatique a donné lieu à la description dans ce mémoire de **nouveaux "objets" de vision** comme le "Modèle de Spécification Fonctionnel de Référence" (SADT) ou la "Boîte Générique du Procédé de vision" (SPEX). Le MCD de Référence de la vision est également un "objet" de vision.

Le cercle spécialiste de la vision se trouve "agrandi" d'un certain nombre **"d'objets"** nouveaux qu'il convient, à défaut de pouvoir les manipuler, de ne pas ignorer. L'ensemble de ces **"objets"** (qui sont des modèles : MCD, SADT, Boîte Fonctionnelle Procédé, Boîte Fonctionnelle Sous-Phase, Programmes) participent à l'encapsulation du savoir faire de la vision au niveau Conception, réalisation du Système d'Information et Fabrication.

La réunion de ces trois activités autour d'un même environnement capable de fournir aux utilisateurs l'ensemble des vues des **objets** qui les composent est à la base de nouveaux projets de recherche tels que CIM-ONE (qui propose un outil orienté objet nommé SHERPA) ou tel que le concept de Design Manufacturing Management System (DMMS) [LOM 92]. Ce concept propose, dans un même environnement CIME, la manipulation de ces **"objets manufacturiers"** et des outils associés à travers leurs *vues conception et fabrication* et pour des *technologies différentes* (vision, usinage, montage, etc...) autour d'une base de données orientée objet (système EMERAUDE, construit sur la norme européenne d'interface PCTE) permettant de gérer la notion d'agrégation et de vue et autorisant en outre une traduction des MCD décrits en formalisme

Entité/Relation vers le modèle objet par l'application de règles de passage (cf. figure 5.1.).

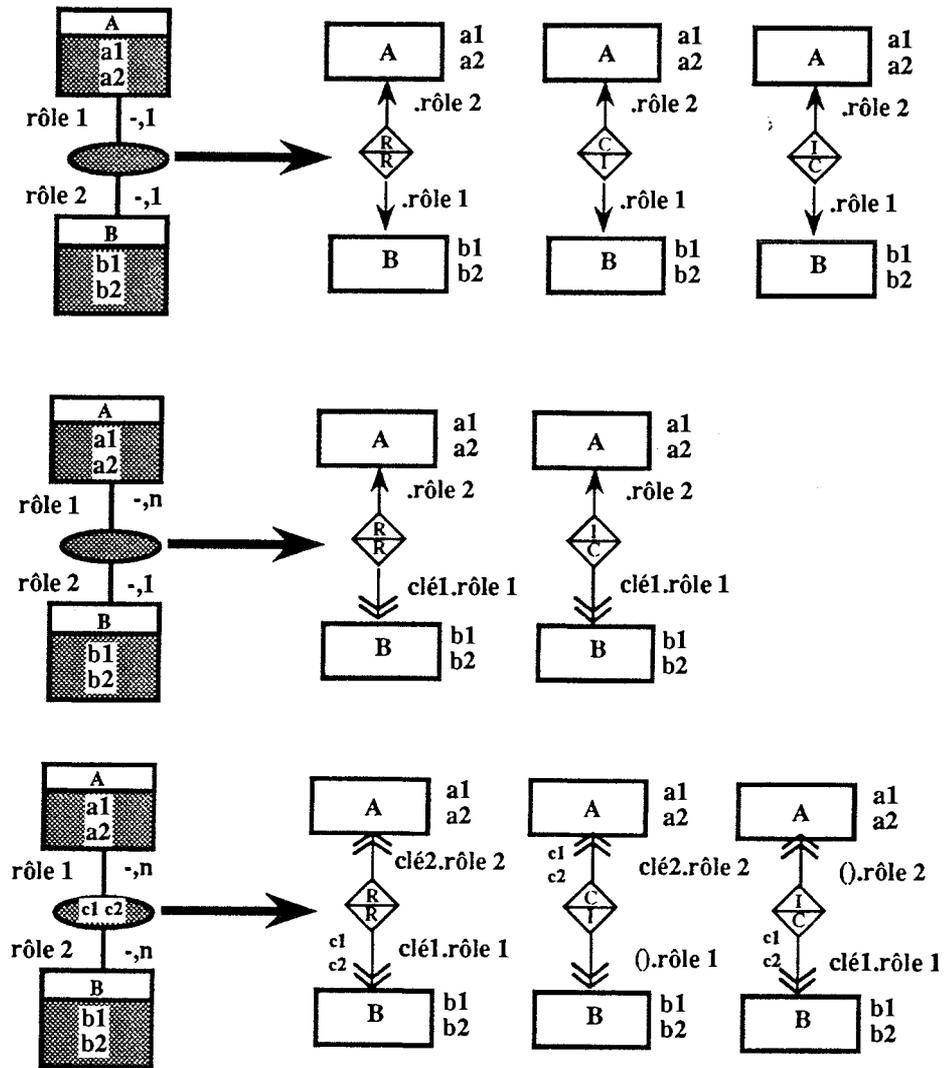


figure 5.1. Règles de passage formalisme Entité/Association étendu au modèle objet dans EMERAUDE

Ce mémoire doit donc contribuer à la spécification d'un **Poste de Travail de Vision** dans un cadre DMMS (cf. figure 5.2.) : nous avons détaillé l'ensemble des informations (MCD) et des activités (SADT, BF) propres à la réalisation de la vision, depuis la conception des processus jusqu'à la fabrication.

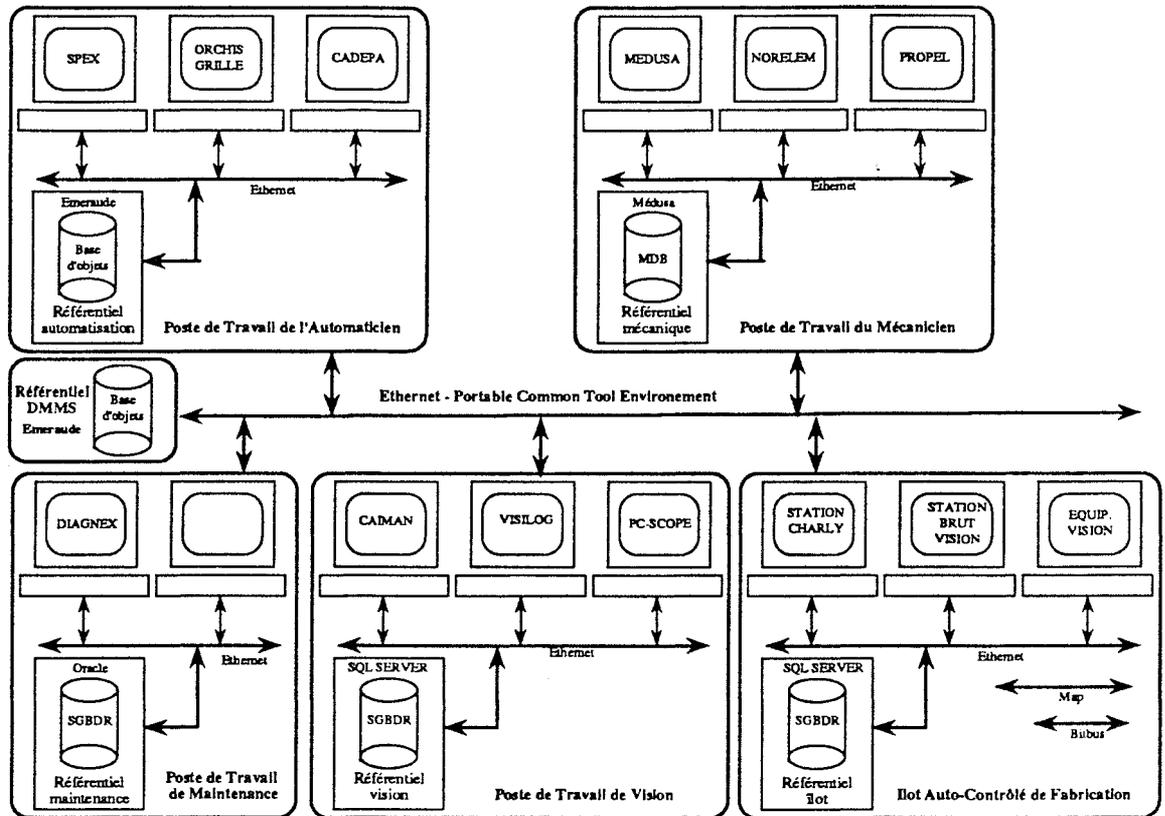


figure 5.2. Architecture DMMS intégrant un Poste de Travail de Vision

## **bibliographie**

- [ADE 84] ADEPA : "Apport de la modélisation de la partie opérative aux différentes étapes de la vie d'un automatisme", synthèse des travaux de la commission, 1984
- [ADI 86] G.MOREL, J. RICHARD, F. LEPAGE : "Robotisation d'un flot de production de pièces mécaniques", convention de recherche ADI - Université de Nancy I 83/742
- [ALA 86] ALANCHE P, LHOTE P, MOREL G, ROESCH M, SALIM M, SALVI PH : "Application de la modélisation de partie opérative à la structuration de la commande", congrès AFCET "Méthodes et outils modernes de conception et d'exploitation de la commande de procédés discontinus complexes" - Montpellier, 1986
- [ANS 91] B. ANSELMETTI, F. VILLENEUVE : "Gamme automatique, deux méthodes génératives de conception de processus d'usinage avec expertise répartie", 23<sup>ème</sup> CIRP, Séminaire international sur les systèmes de production, 6-7 juin 1991, Nancy, France
- [ANT 87] ANTONY RICHARD : Center for signals warfare US army communication electronics command Warrenton, VA 22186-5100 "Spatial reasoning using an object-oriented spatial DBMS", conference paper: spatial reasoning and multi-sensor fusion: proceedings of the 1987 workshop, ST. Charles, IL, USA 5-7 oct 1987, p 42-51, 4 ref, publisher : Morgan Kaufmann Los Altos, CA, USA
- [AYA 83] N. AYACHE : "Un système de vision bidimensionnel en robotique industrielle", thèse de l'université de Paris Sud, centre d'Orsay, 1983
- [BAR 91] J.P. BARDINET : "FIP, un vrai réseau de terrain ouvert : état du projet et perspectives", Automation 91, p 91-101, conférence du 16 mai 1991
- [BEA 89] OLIVIER COSTA DE BEAUREGARD : "Information et entropie" article paru dans l'Encyclopédie Universalis, édition 1989, Corpus 17, page 904

- [BEL 89] A.BELHIMEUR : "Contribution à l'étude d'une méthode de conception des automatismes et des systèmes de conduite des processus industriels", thèse de l'université des sciences et techniques de Lille Flandres Artois, 1989
- [BES 88] J.E. BESANCON : "La vision par ordinateur en deux et trois dimensions", Eyrolles, 1988
- [BOL 86] C. BOLLES, P. HORAUD : "3DPO: a three dimensional Part Orientation System", The International Journal of Robotics Research, vol 5, n°3, Fall 1986
- [BRO 89] J. BROLIO : "ISR: a database for symbolic processing in computer vision", COMPUTER december 1989
- [CAL 89] JP. CALVEZ : "Spécification et conception des systèmes, une méthodologie : MCSE", Editions Masson, Paris, 1990
- [CEG 91] CEGELEC : "AGA, L'apport des normes et des standards", Automation 91, CNIT PARIS, 14 mai 1991
- [CHA 89] T.C. CHANG, D.C. ANDERSON, O.R. MITCHELL : "QTC, an integrated design/manufacturing/inspection system for prismatic parts", ASME, computer in engineering, july 88, San Fransisco, California, editeurs : V.A. JIPNIS, E.M. PATTON, 1989
- [CHE 90] D. CHEN, B. VALLESPER, M. ZANETTIN : "Architecture globale CIM : application dans le projet ESPRIT-IMPACS" , actes de CIM/90 - p 243-250 - 12/14 juin 1990 Bordeaux
- [CIA 87] CIAME-AFCET : "les capteurs intelligents, réflexion des utilisateurs, note de synthèse du livre blanc, CIAME 1987
- [CIC 87] J. CICCOTELLI, R. HUSSON : "Identification et codage d'objets en un passage par analyse d'image", Revue TS, Vol 4, n° 1, 1987, p 33-43

- [CIM 90] Colloque international, "productique et intégration, cim : integration aspects", 12-14 juin 1990 Bordeaux
- [CON 89] Christopher C. McCONNEL : "Construct for cooperative image understanding environment", Image understanding workshop proceedings, Los Angeles, USA, 23-25, 1987, p. 497-506 vol 2, 3 ref, publisher : Morgan Kaufmann, Los Altos, USA
- [COR 89] F. CORBIER : "Modélisation et émulation de la partie opérative pour la recette en plateforme d'équipement automatisés", thèse de l'université de Nancy I Option MAE - 30 juin 1989
- [COU 91] JF. COUTURIER : " Auto-contrôle d'une cellule flexible d'usinage, étude et réalisation du système d'information", thèse de doctorat de l'université de Nancy I, LACN, dec 91
- [DEF 86] J. DEFRENNE : "Modélisation de la partie opérative; impact sur la sécurité et la maintenance des systèmes à évolution séquentielle", thèse de docteur ès sciences, Lille, 1986
- [DEL 87] DELMAIRE, GUARESCHI, STAROSWIEKI : "Aide à la conduite, aide à la maintenance, aide à la gestion technique", conférence technique Automation 1987, Paris
- [DIA 89] S. DE LA VALLEE POUSSIN : "Le projet DIAS (Distributed Intelligence Actuators and Sensors)", journée d'étude "FIP et les constituants intelligents d'automatismes : capteurs, actionneurs, régulateurs et automates", 29-30 nov 1989, MRT, Paris
- [DOI 92] C. DOITEAU, JF. COUTURIER, J. RICHARD, E. BAJIC : "Integration and Control Loop of a Flexible Manufacturing Cell", CRAN/CNRS URA 821, conf "Computer Application in Production and Engineering : Integration Aspects" Bordeaux, France, 10-12 Sep 1991

- [DOU 90] G. DOUMEINGTS : "Méthodes pour concevoir et spécifier les systèmes de production", CIM 90, 12-14 juin 1990, Bordeaux
- [DOU 91] G. DOUMEINGTS, B. VALLESPER : "Techniques de modélisation pour la productique", actes congrès CIRP Nancy, 1991
- [DUP 88] G. DUPRAT, C. LUGAND : "élaboration d'une méthode permettant le choix d'un ensemble de vision adapté à une application donnée", rapport de stage en entreprise, Peugeot Sochaux, DESS Productique et Automatique Industrielle ISIAL Nancy, fac. des Sciences
- [ELA 88] M. ELARBI-BOUDHIR, M. DUFAUT, R. HUSSON : "Guidage de robot mobile par détection de bord de route, étude de la phase initiale de navigation", Revue d'Automatique et de Productique Appliquée RAPA, Vol 3, n°2, 1990 pp 25-39
- [FRA 87] J.P FRACHET : "Une introduction au Génie Automatique : faisabilité d'une chaîne d'outils CAO pour la conception et l'exploitation des machines automatiques industrielles", thèse d'état, Université de Nancy I, 1987
- [GAC 90] R.GACHES, B.QUERENET, P.VIOLLET, F.B VERNADAT: "CIM-OSA : une architecture ouverte pour la productique", actes de CIM/90 - p 227-234 - 12/14 juin 1990 Bordeaux
- [GAL 86] GALASCI (nom collectif) : "Les systèmes d'information - analyse et conception" ed. BORDAS, 1986
- [GEM ad] ADEPA : "Notice de présentation et d'utilisation du GEMMA, Guide d'Etude des Modes de Marche et d'Arrêt", source ADEPA
- [GOO 89] A.M. GOODMAN, R.M. HARALICK, L.G. SHAPIRO : "Knowledge-based computer vision. Integrated programming language and data management system design", COMPUTER, december 1989
- [GRO 89] W.I. GROSKY, R. MEHROTRA : "Image database management", COMPUTER december 1989

- [HAN 87] C. HANSEN, T.HENDERSON : "CADG-Based Computer Vision", IEEE Workshop on computer vision, nov 87 Miami Beach, FL, éditeur IEEE Computer Society, p 100-105
- [HEI 91] MANUEL V. HEITOR : "Advanced sensor systems for the application of CIME technologies in the process industry : a review", Congrès : "CIM in the Process Industry", Proceedings of the Workshop, 14-15 march 1991, Athens, Greece
- [HOR 89] O. HORN, D. WOLF R. HUSSON : "Filtrage de Kalman adaptatif appliqué à la poursuite de cible", Congrès AIPAC 89, 3-5 juillet 1989, Nancy France
- [IGL 89] IGL technology : "SADT, un langage pour communiquer", Eyrolles 1989
- [III 89] Institut Informatique Industrielle : "plateforme générique de développements de procédés intégrant des systèmes de vision industrielle", projet Anvar n° A 8801005 E 007 0
- [ISM 83] ISMCM, VALORIS : "Modélisation, simulation et évaluation de la partie opérative d'un système automatisé", rapport scientifique de fin de contrat, agence de l'informatique, convention de recherche n° 81/545, mai 1983
- [IUN 91] B. IUNG, P. LHOSTE, G. MOREL, M. ROECH : "Functionnal modelling of an Intelligent Actuator : applicable to an ON/OFF or modulating electrical valve", Workshop ESPRIT-CIM; Athènes june 91
- [JAG 89] HV. JAGADISH : "An object model for image recognition", COMPUTER, december 1989
- [JEA 87] R. JEANSOULIN : "L'archivage des images : des fichier aux bases de données", Courrier du CNRS, N° spécial Imagerie Scientifique, 66-67-68, juin 1987
- [JOR 90] P. JORION : "Principes des systèmes intelligents", Masson, Paris, 1990

- [KAK 87] A.C. KAK, A.J. VAYDA, R.L. CROMVELL, W.Y. KIM, C.H. CHEN : "Knowledge-Based-Robotics", Congrès : "Robotics and Automation", int conf papers - Raleigh - mars 87, éditeur : IEEE, p 637-646
- [KAN 89] M. KANUMURY, J. SHAH, T.C. CHANG : "An automatic process planning system for quick turnaround cell, an integrated CAD and CAM system", ASME, USA, Japan symposium on flexible automation, july 88
- [KAT 87] KATSUCHI IKEUCHI : "Generating an interpretation tree from CAD model for 3D object recognition in bin picking tasks", International Journal of Computer Vision, p 145-165 (1987)
- [KAT 88] KATSUCHI IKEUCHI, TAKEO KANADE : "Automatic generation of object recognition programs", proceedings of the IEEE, vol 76, n°8, august 1988
- [LAB 87] Henri LABORIT : "Dieu ne joue pas au dés" éditions Grasset & Fasquelle, 1987.
- [LEV 88] E. LEVRAT, P. BERTHOME, B. HEIT, J. BREMONT, M. LAMOTTE : "Asservissement de position piloté par un capteur visuel", Revue d'Automatique et de Productique Appliquée RAPA, Vol 1, n° 2, 1988, p 65-73
- [LEV 89] E. LEVRAT, J. BREMONT, B. HEIT, G. DUBOIS : "Edge detection using the fuzzy theory", AIPAC 89, IFAC SYMPOSIUM, P II-171-174, 3-5 juillet 1989, Nancy, France
- [LHO 85] P.LHOSTE : "Exploitation des systèmes automatisés, EXAO: proposition d'une approche méthodologique et d'outil d'assistance", thèse de l'université de Nancy I, 1985
- [LOM 90] LOMBARD MURIEL : "Proposition d'un Modèle Conceptuel de Données pour îlot ouvert de fabrication mécanique", DEA Production Automatisé, CRAN/LACN, Université de NancyI, 1990

- [LOM 92] M. LOMBARD, B. ROCHOTTE, F. MAYER, G. MOREL : "Design Management and Manufacturing Concept for flexible manufacturing system", conf. MATADOR, 92, Manchester, (à paraître)
- [MAS 89] MASAKAZU EJIRI : "Knowledge-based approaches to practical image processing", International workshop on industrial applications of machine intelligence and vision (miv 89), Tokyo, april 10-12, 1989; publisher : IEEE, New-York,USA
- [MAT 86] L. MATHIEU : "Etude des limites des paramètres de coupe en vue d'un choix automatique des outils et des conditions de coupe", thèse productique de l'Ecole Centrale de Paris, 24 avril 86
- [MEG 89] MEURISE-MEGA, MEGA - Manuel de référence - 1989, ESPACE - MICRO - Manuel utilisateur - 1989, MEGA - REALISATION - Manuel de référence - 1989, GAMMA International - PARIS
- [MEY 86] B. MEYER : "Génie Logiciel", techniques de l'ingénieur, h2050
- [MEZ 88] G. MEZIN : "La vision par ordinateur", Editions Hermes, technologies de pointe
- [MOR 88] G. MOREL, M. ROESCH, M. VERON : "Génie Productique, Génie X", congrès Afcet Automatique, Grenoble, p320-p330
- [PRI 90] Projet PRIMECA, Pôle de Ressources Informatiques pour la MECANIQUE, Serge TICHKIEWITCH, chef de projet
- [MOR 89] G.MOREL : "Automatisation intégrée d'un flot de fabrication manufacturière", communication GRECO CNRS " systèmes à événements discrets" groupe GT4, 12 janvier 1989 Paris
- [MOR 89a] G. MOREL, M. ROESCH, B. IUNG : "Les outils de X.A.O. dans le cycle de vie de l'automatisation", quelle CAO pour les systèmes automatisés de production. Journée d'études. SEE-club18, 18 janv 1989, Gif sur Yvette

- [MOR 90] G.MOREL : "Automatisation intégrée d'un îlot de fabrication manufacturière", CIM 90, 12-14 juin 1990, Bordeaux
- [MOR 92] G. MOREL : "Automatisation et Ingénierie Intégrées de Production", habilitation d'encadrement doctorale et de recherche, 92
- [MRT 91] CRAN, LAB, LRII contrat de recherche MRT n°) 88 p 0718 : "Auto-contrôle de cellule flexible d'usinage, intégration dans un ensemble CFAO", rapport de fin de contrat
- [MRT 91] ROCHOTTE B. MOREL G. : "Auto-contrôle de cellules flexible d'usinage, Intégration dans un ensemble CFAO". Contrat de recherche productique Robotique 88 Ministère de le recherche et de la technologie, N°86.p.0719. Rapport scientifique final. Université de Nancy I. Mai 1991. "Aspect vision"
- [MUN 88] F. MUNERATO : "Robotisation intégrée d'un îlot de production manufacturière: aspects contrôle-commande et communication", thèse de doctorat de l'université de Nancy I, 18 avril 1988
- [NAD 91] A. NADIF, B. MUTEL, R. BALLOT : "Application de la technologie de groupe au système de production", 23 ème CIRP, Séminaire international sur les systèmes de production, 6-7 juin 1991, Nancy, France
- [PAN 91] Hervé PANETTO : "Une contribution au génie automatique : le prototypage des machines et systèmes automatisés de production", thèse de doctorat de l'université de Nancy I, LACN, janvier 91
- [PAR 88a] H. D. PARK, O. ROBERT MITCHELL : "Automated computer vision inspected system for quick turnaroud manufacturing", conférence : automated inspection and high speed vision architecture II, Cambridge, USA, 10-11 nov 1988, proc of the SPIE, the international society for optical engineering, vol 1004, 1989, p.114-25, 16 ref, coden PSISDG, ISSN: 0277-786X, USA

- [PAR 88b] H. D. PARK, O. ROBERT MITCHELL : "CAD based planning and execution of inspection", conférence : Computer vision and patern recognition, proc of the IEEE, june 88
- [PIZ 89] ARTURO PIZANO : "Specification of spatial integrity constraints in pictorial database", COMPUTER december 1989
- [PRU 86] PRUNET F., CASALOT C., LLORCA P., DECCHENAUX G. : "De l'automatisme simple à l'atelier flexible avec le grafcet", congrès AFCET automatique, mars 1986, Montpellier
- [PTA 87] PTA : "poste de travail pour l'automatisation", synthèse MRES, RNUR, MICHELIN, ADEPA, PSA, SGN-ISMCM, LACN, LAG, LAMM
- [PUE 87] C. PUECH : "Synthèse d'image", Courrier du CNRS, N° spécial Imagerie Scientifique, 66-67-68, juin 1987
- [QIA 89] QUALITY IN AUTOMATION : "Process report of the quality in automation project for FY88", NISTIR 89, 4045 US, departement of commerce, 1989
- [QIN 88] HU QING, YANG JINGYU : "A method of establishing pattern database for machine vision systems", proceedings of the 1988 IEEE international conference on systems, man, and cybernetics, Beijing and Shenyang, China 8-12 aug 1988, p 486-9 vol.1, publisher: int. acad. publishers. Beijing, China
- [RIC 91] J. RICHARD, F. LEPAGE, G. MOREL : "Computer/ programmable control of a flexible manufacturing cell", international series on "microprocessor-based systems engineering", vol 6, "microprocessors in robotic and manufacturing systems", Kluwer Academic Publishers, ISBN 0-7923-0780-1
- [RIC 88] J.RICHARD : "Auto-contrôle d'une cellule flexible d'usinage; intégration dans un ensemble CFAO" Convention MRES 1988

- [ROB 88] M. ROBOAM : "Modèles de référence et intégration des méthodes d'analyse pour la conception des systèmes de production", thèse de l'université de Bordeaux 1988
- [ROC 88] B. ROCHOTTE : "Méthodes structurelles pour la reconnaissance de forme; application de la méthode de prédiction vérification à la reconnaissance et la localisation d'objet, en vue de l'intégration d'un système de vision dans une cellule flexible de fabrication." DEA de production automatisée de l'université de Nancy I, LACN septembre 88
- [SFA 89] A. SFALCIN : "Fiabilité, disponibilité et sécurité des installations intégrant des filtres de comportement", colloque annuel du club FIABEX, Paris, 22-23 nov 89
- [SHE 90] D. RIEUX, V. FAVIER, P. JEAN, B. DAVID, G.T. NGUYEN : "SHERPA : un support d'intégration pour le processus CDEF", actes de CIM/90 - p 451 - 12/14 juin 1990 Bordeaux
- [SHI 87] F.Y SHIH, O.R MITCHELL : "Skeletonization and distance transformation by grayscale morphology", Proc SPIE Symposium on automated inspection and high speed vision architecture, Cambridge, MA, 1987
- [SOU 83] V. SOUVIGNIER : "PVV, un système d'interprétation d'images par prédiction et vérification", thèse de l'INP de Grenoble, 1983
- [STA 89] M.STAROSWIECKI : "Les actionneurs intelligents", journée d'études "FIP et les constituants intelligents d'automatismes: capteurs, actionneurs, régulateurs de automates", 29-30 nov 1989, MRT, Paris
- [STA 90] M. STAROSWIECKI : "Les actionneurs intelligents", Revue générale de l'électricité, n°7, juillet 1990
- [SUT 91] S. SUTER : "Elaboration d'un cahier des charges à partir d'une approche système", dess PAI de l'université de Nancy I
- [TAB 86] TABOURIER YVES : "De l'autre coté de Merise - Systèmes d'information et modèles d'entreprise" Edition d'organisation Paris 1986

- [TIM 88] EDITIONS TIME-LIFE : "L'intelligence artificielle", Amsterdam, 1988
- [TIX 89] JM. TIXADOR : "Une contribution au génie automatique: la spécification exécutable des machines et systèmes automatisés de production", thèse de l'université de Nancy I, 1989
- [TSA 87] J.P TSANG : "Planification par combinaison de plans; application à la génération de gammes d'usinage", thèse de l'INP de grenoble, 1987
- [VOG 86] R. VOGRIG : "Flexible manufacturing shop operation", 18 th CIRPMFSS, Stuttgart, F.R. of germany, june 4-5 1986
- [VOG 89] C. VOGEL : "Génie cognitif", collection science et technique, édition Masson 1980
- [VOG 90] R. VOGRIG : "Intégration d'un capteur multi-fonctionnel de vision dans un flot de fabrication de pièces mécaniques: aspects méthodologique et réalisationnel", thèse de doctorat de l'université de Nancy I, LACN, juillet 1990
- [WEN 87] P.L. WENDEL : "Architecture des systèmes de traitement et de synthèse d'images", Courrier du CNRS, N° spécial Imagerie Scientifique, 66-67-68, juin 1987
- [WIN 88] Patrick Henry WINSTON : "intelligence artificielle", InterEdition, 1988
- [WOL 90] D. WOLF, R. HUSSON : "Inspection de surfaces planes à dominante spéculaire", Revue d'Automatique et de Productique Appliquée RAPA, Vol 3, n° 2, 1990, p 25-39

## Liste des figures

### Figures du chapitre I

figure 1.1. Modèle CAMI.....	3
figure 1.2. Modèle NIST .....	4
figure 1.3. Modèle Conceptuel de Référence Informationnel de l'Entreprise [MOR89]..	5
figure 1.4. Le concept de CMMS [DIA 89] .....	7
figure 1.5. Le cadre de modélisation IMPACS [CHE 90] .....	11
figure 1.6. Le cadre de modélisation CIM-OSA [GAC 90] .....	12
figure 1.7. La situation de nos travaux dans le modèle CIM-OSA.....	14
figure 1.8. Le formalisme Entité/Relation étendu .....	16
figure 1.9. Concepts de MEGA .....	17
figure 1.10. Décomposition d'un processus.....	19
figure 1.11. Processus de décomposition d'une fonction de contrôle .....	20
figure 1.12. Grille associée à une fonction de contrôle-commande .....	21
figure 1.13. Vue interne de la boîte fonctionnelle : commander une lampe.....	23

### Figures du chapitre II

figure 2.1. Processus élémentaire d'usinage .....	3
figure 2.2. Processus élémentaire de contrôle sans contact .....	3
figure 2.3. Extrait du MCD de référence de vision : MONTAGE SOURCE .....	5
figure 2.4. Extrait du MCD de référence de vision : MONTAGE SCENE .....	6
figure 2.5. Extrait du MCD de référence de vision : MONTAGE CAPTEUR .....	8
figure 2.6. Extrait du MCD de référence de vision : MONTAGE OPTIQUE .....	10
figure 2.7. Relation entre objets dans un modèle du monde selon [GOO 89] .....	13
figure 2.8. Principe de base de compréhension .....	18
figure 2.9. Extrait du MCD de référence de vision : représentation des traitements .....	23
figure 2.10a. Extrait du MCD de référence de vision : représentation du monde visuel..	24
figure 2.10b. Extrait du MCD de référence de vision : représentation du monde réel dans la cellule .....	24
figure 2.11. Extrait du MCD de référence de vision : les exécutable et les données utilisés par les traitements .....	25
figure 2.12. Extrait du MCD de référence de vision : les résultats d'une opération de vision .....	27
figure 2.13. Extrait du MCD de référence de vision : Le système de vision .....	28
figure 2.14. La pièce NASA .....	30
figure 2.15. Fichier de commande et fichier de configuration de la carte EDGE : contour binaire .....	31
figure 2.16. Fichier d'étalonnage .....	32
figure 2.17. Fichier modèle d'objet pour la prédiction vérification.....	33
figure 2.18. Fichier modèle d'objet pour la mesure d'un perçage.....	34
figure 2.19. MCD Partiel : les résultats d'une opération de vision .....	35
figure 2.20. MOD Particulier : les résultats d'une opération de vision dans la base .....	36

figure 2.21. MLD Particulier : les résultats d'une opération de vision dans la base.....	37
figure 2.22. Génération du code SQL à partir du MLD .....	38
figure 2.23. Requêtes SQL dans un programme vision .....	39

### Figures du chapitre III

figure 3.1. Architecture fonctionnelle des pilotages cellule et station [RIC 91].....	4
figure 3.2. SADT de Spécifications Fonctionnelles de Référence : réaliser une opération de vision .....	12
figure 3.3. Extrait du MCD de référence de vision : opération de vision.....	14
figure 3.4. Architecture physique de la cellule de la maquette MRT.....	20
figure 3.5. SADT de Spécifications Fonctionnelles Particulier : réaliser un contrôle de perçage sur une fraiseuse CHARLY-ROBOT.....	23
figure 3.6. SADT de Spécifications/Conception : réaliser un contrôle de perçage sur une fraiseuse CHARLY-ROBOT.....	24
figure 3.7. GRILLE : réaliser un contrôle de perçage sur une fraiseuse CHARLY- ROBOT.....	25
figure 3.8. Modèle de Filtre prototypé sur SPEX : modules commander, surveiller la lampe .....	27
figure 3.9. Extrait de programme en PLM51 de la commande des lampes sur un noeud BITBUS .....	28
figure 3.10. Extrait de programme en code NUM760 : Positionner Caméra, Dégager Caméra.....	29
figure 3.11. Extrait du programme de prise d'image .....	30
figure 3.12. MCD Partiel : opération de vision pour un contrôle d'une pièce .....	33
figure 3.13. MOD Particulier : opération de vision dans la base de donnée de la cellule .....	34
figure 3.14. MLD Particulier : opération de vision dans la base de données de la cellule .....	35
figure 3.15. Code SQL : opération de vision.....	36

### Figures du chapitre IV

figure 4.1. Situation des méthodes de conception des processus dans le trièdre méthodologique.....	5
figure 4.2. Extrait du MCD de référence de vision : sémantique du système de vision .....	10
figure 4.3. Extrait du MCD de vision : le choix des moyens .....	11
figure 4.4. Fonctionnement d'un système de vision.....	24
figure 4.5. Extrait du MCD de vision : les données nécessaires à la conception des connaissances consommées par un système de vision.....	26
figure 4.6. Définition par entité du gammiste de la pièce NASA .....	32
figure 4.7. Gamme d'usinage de la pièce NASA .....	33
Figure 4.8. Application : le niveau station et le niveau équipement .....	42
figure 4.9. Commande du procédé de vision sur le CHARLY-ROBOT.....	44
figure 4.10. Commande du procédé d'usinage sur le CHARLY-ROBOT .....	45
figure 4.11. Commande de la SOUS-PHASE.....	47

### Figures de la conclusion

<b>figure 5.1. Règles de passage formalisme Entité/Association étendu au modèle objet dans EMERAUDE .....</b>	<b>3</b>
<b>figure 5.2. Architecture DMMS intégrant un Poste de Travail de Vision .....</b>	<b>4</b>

## Abréviations

<b>AMICE</b>	European Computer Integrated Manufacturing Architecture
<b>API</b>	Automate Programmable Industriel
<b>B-REP</b>	Boundary Representation
<b>BF</b>	Boite Fonctionnelle (SPEX)
<b>BITBUS</b>	réseau de terrain
<b>CAD</b>	Computer Aided Design
<b>CAM</b>	Computer Aided Manufacturing
<b>CAO</b>	Conception Assitée par Odinateur
<b>CDV</b>	Cycle De Vie
<b>CFAO</b>	Conception et Fabrication Assités par Ordinateur
<b>CIAME</b>	Commission Industrie-Administration pour la Mesure
<b>CIM</b>	Computer Integrated Manufacturing
<b>CIM-OSA</b>	Computer Integrated Manufacturing Open System Architecture
<b>CIME</b>	Computer Integrated Manufacturing Engineering
<b>CMMS</b>	Control Manufacturing Management System
<b>CN</b>	Commande Numérique
<b>CRAN</b>	Centre de Recherche en Automatique de Nancy
<b>CSG</b>	Constructive Solid Geometry
<b>DF</b>	Diagramme Fonctionnel (SPEX)
<b>DIAS</b>	Distributed Intelligent Actuator and Sensor
<b>DMMS</b>	Design Manufacturing Management System
<b>E/R</b>	Entity-Relationship (Entité/Relation)
<b>EDB</b>	Environmental DataBase
<b>EPO</b>	Elément de Partie Opérative
<b>ICAM</b>	Integrated Computer Aided Manufacturing (modèle d'entreprise)
<b>IDEF</b>	Formalismes du projet ICAM
<b>IDEF0</b>	modèle fonctionnel
<b>IDEF1</b>	modèle de données
<b>IDEF2</b>	modèle de flux de produit (simulation)

<b>ISR</b>	Intermediate Symbolic Representation
<b>LAB</b>	Laboratoire d'Automatique de Besançon
<b>LACN</b>	Laboratoire d'Automatique et de Commande Numérique
<b>LRII</b>	Laboratoire de Robotique et d'Informatique Industrielle de Nantes
<b>LTDB</b>	Long Terme DataBase
<b>MAP</b>	Manufacturing Automation Protocol; réseau local industriel
<b>MCD</b>	Modèle Conceptuel de Données
<b>MERISE</b>	méthode d'analyse et de conception des systèmes d'information
<b>MFA</b>	Module Fonctionnel d'Automatisme
<b>MLD</b>	Modèle Logique de Données
<b>MMS</b>	Manufacturing Message Specification
<b>MOD</b>	Modèle Organisationnel de Données
<b>MPD</b>	Modèle Physique de Données
<b>MSAP</b>	Machines et Systèmes Automatisés de Production
<b>NIAM</b>	Modèle de Données de Nijssen, Relationnel Binaire
<b>NIST</b>	National Institute of Standard and Technology (modèle d'entreprise)
<b>PC</b>	Personal Computer
<b>PC</b>	Partie Commande
<b>PCTE</b>	Portable Common Tool Environment (standard d'échange)
<b>PDB</b>	Programmers DataBase
<b>PEU</b>	Processus Élémentaire d'Usinage
<b>PO</b>	Partie Opérative
<b>PSDB</b>	Perceptual Structure DataBase
<b>PTA</b>	Poste de Travail pour l'Automaticien
<b>PTV</b>	Poste de Travail pour la Visionique
<b>QTC</b>	Quick Turnaround Cell manufacturing
<b>SADT</b>	Structured Analysis and Design Techniques
<b>SGBD</b>	Système de Gestion de Base de Données
<b>SII</b>	Système d'Information Industriel
<b>SIP</b>	Système Intégré de Production
<b>SPEX</b>	SPécification EXécutable
<b>SQL</b>	Standard Query Language
<b>TLC</b>	Thin Line Code
<b>TQM</b>	Transformation-Qualité-Manutention

**VMD**      **Virtual Manufacturing Device (struture de données dans MMS)**

# UNIVERSITE DE NANCY I

NOM DE L'ETUDIANT : Monsieur ROCHOTTE Bruno

NATURE DE LA THESE : DOCTORAT DE L'UNIVERSITE DE NANCY I  
PRODUCTION AUTOMATISEE



VU, APPROUVE ET PERMIS D'IMPRIMER

NANCY, le 22 JAN. 1992 n° 35

LE PRESIDENT DE L'UNIVERSITE DE NANCY I



**RESUME** : ce mémoire de recherche fait une proposition d'architecture physique d'Ilot Auto-Contrôlée de Fabrication Manufacturière intégrant un système de vision **multi-utilisateurs, multi-fonctions, intégré, ouvert et auto-contrôlé**. Sur cette base physique, nous proposons à la fois des **Méthodologies de Conception** et des **Modèles de Référence** appliqués aux points de vue **informationnel** et **fonctionnel** de la visionique considérée comme un domaine intégré dans le Système de Production. Le point de vue Informationnel a pour but la conception du Système d'Information Industriel; nous utilisons le formalisme **Entité/Association étendu** et la méthode **MERISE**; nous proposons une première approche de l'ensemble des Modèles de Référence Conceptuels de Données de la Visionique. Le point de vue Fonctionnel a pour but la conception du Comportement (Contrôle-Commande) de la Machine Vision dans l'ilot; nous utilisons les **outils et méthodes du Génie Automatique** (SADT, SPEX...). L'ensemble des modèles proposé (informationnels et fonctionnels) contribue à la formalisation du **savoir-faire** (sémantique) métier de la visionique, utile dans une optique de réutilisation (reconception).

**MOTS CLES** : Visionique, CIM, Capteur Intelligent, Méthodologie, Modèle de Référence Fonctionnel, Modèle de Référence Informationnel, Design Management and Manufacturing System, Ingénierie