

# Structure arborescente d'inférences floues pour l'intégration de connaissances métiers

## Arborescent structure based on fuzzy inferences for the integration of expert knowledge

C. Mazaud

V. Bombardier

P. Lhoste

Centre de Recherche en Automatique de Nancy,  
CNRS, UMR 7039

Faculté des Sciences - Bd des Aiguillettes –  
BP 239 – 54506 Vandoeuvre les Nancy Cedex

Vincent.Bombardier@cran.uhp-nancy.fr

Pascal.Lhoste@cran.uhp-nancy.fr

### Résumé

*Cet article expose une méthodologie d'intégration de connaissances pour l'amélioration d'un système de reconnaissance de défauts par vision sur des planches de bois.*

*Nous situons le problème de vision qui est à la base de cette étude, puis nous explicitons les connaissances métier nécessaires, aussi bien dans le domaine du métier du bois que dans le domaine de la vision. Nous utilisons pour cela un modèle symbolique basé sur la méthode NIAM/ORM, formalisant ces connaissances métier à partir de leur expression en langage naturel.*

*Pour améliorer le système de reconnaissance de formes existant, nous proposons une méthodologie d'intégration des connaissances métier pour construire un modèle numérique. Ce modèle est créé selon une structure arborescente où chaque noeud consiste en un moteur d'inférence à base de règles linguistiques floues. Les résultats obtenus prouvent l'intérêt de cette méthodologie.*

### Mots Clés

Identification de défauts, logique floue, règles linguistiques floues, méthode N.I.A.M., O.R.M., modélisation de connaissances métiers.

### Abstract

*This article exposes a methodology of knowledge's integration for the improvement of a system of defects recognition by vision on wooden boards.*

*We locate the problem of vision which is at the base of this study, then we clarify the expert knowledge, as well in the field of the wood profession as in the field of vision. We use for that a symbolic model based on the NIAM/ORM method, formalizing the expert knowledge expressed in natural language.*

*To improve the system of shapes recognition, we propose a methodology to construct the numeric model. This model is built according to an arborescent structure where each node represents*

*an engine of inference based on fuzzy linguistic rules. The results obtained prove the interest of this methodology.*

### Keywords

Defect identification, fuzzy logic, fuzzy linguistic rules, N.I.A.M. method, O.R.M., modelling of expert knowledge.

### 1 Introduction

L'accroissement du niveau d'Information dont on peut disposer sur un système est essentiel pour l'amélioration de la maîtrise de ce système et des processus qui y sont intégrés (automatisation, maintenance, relations avec l'environnement, ...). Deux sources principales permettent l'accès à cette Information : d'une part, à partir de *connaissances* d'experts humains (experts en procédés, en produits, en normes, ...) qui donnent une information plutôt qualitative sur ce qu'est le système étudié, et, d'autre part, par acquisition de *données* directement sur le système, donnant plutôt une information quantitative sous la forme de mesures. Le rapprochement de ces sources d'informations aussi différentes que complémentaires doit alors contribuer à une modélisation de l'information plus complète et plus cohérente permettant une intégration plus forte des processus composant le système. En ce sens, l'étude proposée cherche à définir une méthodologie de modélisation et d'intégration permettant, d'une part, de transformer la Connaissance en Information, d'autre part, d'extraire l'Information à partir de Données numériques prélevées sur le système et, enfin, d'intégrer ces Informations d'origines différentes en un seul référentiel d'Information Système. Cette méthodologie est appliquée au domaine de la reconnaissance de formes et, en particulier, à celui de l'identification de défauts sur des planches en bois par système de vision industriel.

## 1.1 Système de vision

Les travaux présentés se situent dans le cadre d'une collaboration université – entreprise entre le CRAN et la société luxembourgeoise Luxscan Technologies et concernent l'élaboration d'un système de reconnaissance par vision de défauts sur des planches de bois. C'est sur la ligne de production que s'effectue en temps réel l'analyse des planches dont la vitesse de déplacement peut atteindre 3 à 5 mètres par seconde en considérant un maximum de 200 défauts par mètre. La figure 1 résume le principe du système de vision.

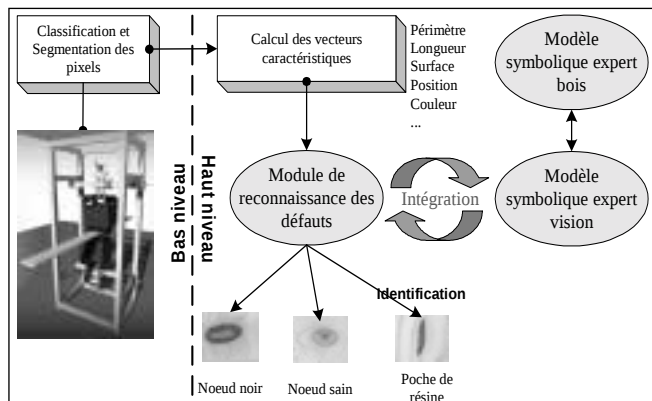


Figure 1 : système de vision

L'acquisition de l'image de la planche de bois s'effectue par des capteurs linéaires qui peuvent être utilisés individuellement ou combinés. Ils sont principalement de deux types :

- couleur fournissant les canaux rouge, vert et bleu,
- lasers qui renvoient une mesure tridimensionnelle de la surface du bois, une mesure de densité et d'orientation, et deux mesures de luminosité en rouge et infrarouge.

Les signaux, délivrés par ces capteurs, sont échantillonnés au rythme de 2000 lignes par seconde et sont quantifiés sur 1024 niveaux. Les images ainsi obtenues sont ensuite pré-traitées et labellisées pour fournir une information ne comportant plus que six classes de pixels. Ces classes définissent la classe de pixels « noir », « brun clair » (correspondant au bois sain), « brun foncé », « orange », « bleu », et « fentes ».

Cette étape de classification est réalisée soit par un classificateur bayésien, soit par un classificateur à seuil dont les paramètres sont réglés empiriquement. La segmentation, réalisée à partir de l'image labellisée, consiste en l'agrégation de pixels connexes n'appartenant pas à la classe de pixels « brun clair » définissant le bois sain. Elle permet donc d'extraire les régions dites « défectueuses » des régions dites « saines ».

Un ensemble de paramètres (surface d'un défaut, orientation, couleur, ...) est calculé sur ces régions pour fournir le vecteur caractéristique utilisé par l'étape d'identification qui a pour objectif de donner un nom au défaut : « noeud noir », « noeud

sain », « noeud cerné », « poche de résine », « bleuissement », « fentes » ...

Les travaux que nous présentons ne concernent que le module de reconnaissance de la partie haut niveau (figure 1).

## 1.2 Méthodologie proposée

Dans le domaine de la reconnaissance de défauts sur des planches de bois, la tendance générale réside en l'utilisation d'un système de vision par caméra pour l'acquisition de l'image et d'un classificateur basé sur des réseaux de neurones supervisés ou non supervisés [11] [12] [15]. Bien que les résultats cités dans les articles semblent satisfaire les auteurs, nous avons choisi de ne pas utiliser de méthode à base de réseaux de neurones pour notre application. En effet, des réseaux de neurones agissent comme une boîte noire à partir de laquelle il est difficile de connaître précisément leur organisation et leur fonctionnement. Dans notre cas, nous désirons connaître le fonctionnement du système afin de pouvoir y intégrer l'information issue des connaissances expertes, et en extraire de l'information pour compléter nos modèles issus des connaissances expertes. C'est donc en partie pour cela que nous avons choisi de ne pas retenir les réseaux de neurones et de nous orienter vers des méthodes plus aisément paramétrables et interprétables comme les systèmes à base de règles.

A ce niveau, l'utilisation d'une méthode mettant en oeuvre les outils de la théorie des ensembles flous et plus particulièrement de la logique floue [4], se justifie par trois raisons.

Premièrement, les défauts à détecter sont intrinsèquement flous, il n'existe pas toujours une frontière stricte entre un défaut et un bois sain. La phase de segmentation, qui ne prend pas en compte cette imprécision, peut fournir un vecteur caractéristique inexact. L'utilisation de la logique floue vise à rendre le système d'identification moins sensible à ces imprécisions.

Deuxièmement, les classes de défauts en sortie ne sont pas toutes disjointes. En effet, la limite entre un « petit noeud noir » et un « grand noeud noir » n'est pas forcément stricte.

Troisièmement, les termes linguistiques utilisés par les experts pour définir les défauts (petit noeud, grand noeud, à peu près circulaire) sont souvent vagues et imprécis. Nous représentons ces termes sous la forme de termes flous caractérisés par leur fonction d'appartenance. Ces termes sont ensuite traités par un mécanisme à base de règles.

Dans le domaine des systèmes qui utilisent une expertise humaine et une expertise vision, la tendance est de construire un système de règles heuristiques [6] [16] [17] décrivant le fonctionnement du système décisionnel. Cependant, bien que les jeux de règles créés semblent fonctionner, aucune précision n'est donnée quant à leur validation par rapport aux

connaissances émises par les experts du domaine concerné. Ce sont donc a priori des processus informels qui permettent cette validation.

Afin de pallier ce problème, nous proposons dans notre méthodologie une approche de validation formelle basée sur l'utilisation de la méthode NIAM [7]. Celle-ci permet, en effet, de transcrire le modèle, traduisant l'information issue de la connaissance de l'expert, en langage naturel compréhensible par l'expert, qui peut ainsi, à travers cette transcription [1], valider le modèle créé.

Pour nos travaux, deux orientations ont donc été retenues pour rapprocher les informations exploitables en identification de défauts sur des planches de bois par vision : celles provenant du système de vision et celles provenant de l'expert « métier du bois ».

D'une part, nous disposons de données numériques sous la forme d'un vecteur qui caractérise le défaut à identifier au travers d'une relation que nous appelons dans la suite de l'article « modèle numérique ». Nous avons choisi d'implémenter ce modèle numérique à l'aide d'un système de règles linguistiques floues afin de faciliter l'interprétation du mécanisme d'identification et de pouvoir en extraire l'information portée par le vecteur caractéristique et la formaliser. Pour obtenir le modèle numérique, nous utilisons la méthode de Nozaki [13] [14], détaillée dans le paragraphe 3.

D'autre part, nous avons la connaissance des défauts à identifier qui est exprimée en langage naturel en des termes métier. Nous avons entrepris une modélisation symbolique des informations portées par la connaissance métier en utilisant la méthode NIAM. Dans la suite de l'article, nous appelons « modèle symbolique » cette modélisation de l'information issue de l'expert « métier du bois ».

L'objectif de la méthodologie proposée, illustrée par la figure 1, est alors de faire collaborer les trois modèles.

La première idée est de pouvoir compléter le modèle numérique grâce au modèle symbolique par des informations non présentes dans les données numériques et ainsi pouvoir, par exemple, modéliser des défauts dont il n'existe pas d'échantillons.

La deuxième idée est de valider le modèle numérique, en comparant les règles générées par l'apprentissage aux connaissances formalisées dans le modèle symbolique. Une différence peut signifier que le lot d'apprentissage n'est pas cohérent avec les défauts que l'on veut reconnaître.

Néanmoins, la mise en correspondance du modèle symbolique et du modèle numérique ne peut se

faire directement car, d'un côté, nous avons des termes du métier du bois et, de l'autre, nous avons des termes issus du domaine du traitement de l'image. Il faut donc un modèle intermédiaire chargé de la « traduction » entre le domaine du bois et le domaine du traitement de l'image.

Dans la suite du document, nous utilisons un seul et même langage de formalisation, NIAM, pour présenter le modèle symbolique des défauts exprimés initialement par l'expert métier en langage naturel, ainsi que le modèle symbolique intermédiaire décrivant les défauts d'un point de vue « procédé de vision », et enfin, le modèle numérique qui représente l'identification des défauts.

## 2 Les modèles symboliques

Le modèle symbolique concerne la formalisation des connaissances expertes dans le domaine du bois ainsi que dans le domaine du procédé d'identification de défauts (système de vision).

La méthode d'analyse retenue pour élaborer ce modèle est la méthode NIAM (Nijssen Information Analysis Method) [7]. Nous utilisons un de ses formalismes dérivés, ORM (Object Role Modeling) [8] [9], pour représenter les résultats de cette analyse. De manière générale, la méthode NIAM distingue le système *objet* représenté par des *LOT* (*Lexical Object Type*), partie de la réalité observable pour laquelle on veut collecter de l'information, et le système *d'abstraction* représenté par des *NOLOT* (*NO Lexical Object Type*), modèle symbolique du système *objet*, constitué de classes d'objets, de classes de règles.

Cette méthode présente un intérêt significatif car elle repose directement sur une expression de faits en langage naturel. Le modèle d'informations résultant peut aussi être soumis à validation par l'expert sous une forme compréhensible, c'est à dire en Langage Naturel Binaire (LNB), transcription du modèle d'information appelée aussi « paraphrasage » [1]. Il est possible de compléter progressivement le modèle en posant à l'expert des questions précises induites par la méthode (par exemple : un défaut possède-t-il une ou plusieurs couleurs) et pouvant ainsi introduire dans le modèle les notions de contraintes d'état ou de transition.

De plus, un modèle NIAM/ORM ainsi obtenu a la capacité d'être dérivé vers des applications informatiques (génération de structures de bases de données, prototypage d'interfaces, ...), et donne la possibilité de générer une base d'informations exploitable à destination des utilisateurs.

### 2.1 Le modèle symbolique de l'expert métier du bois

Afin de créer ce modèle, on demande à l'expert du domaine du bois de rédiger une liste, la plus complète possible, en langage naturel, des différents défauts que l'on rencontre sur le type d'essence manipulée. Il intègre à cette liste des connaissances sur

l'environnement et le contexte des défauts (par exemple, un défaut se trouve en général à proximité d'un autre). Cette expression est suffisante pour reconnaître, à l'œil nu, n'importe quel défaut énoncé sur un échantillon de bois.

Voici un exemple qui permettra d'illustrer ces propos. L'expert bois décrit un défaut de la manière suivante :

« La moelle est un défaut de forme allongée, généralement au centre de la planche. Elle suit la fibre du bois et est de couleur brune ».

D'après la méthode NIAM, on décompose ces phrases brutes en phrases élémentaires non ambiguës. Une phrase élémentaire a une structure simple de type < sujet >, < verbe >, < complément >. L'ensemble des phrases élémentaires issues de la décomposition d'une phrase en langage naturel doit porter la même signification que la phrase initiale. Dans notre exemple, nous obtenons les phrases élémentaires qui suivent.

*Le défaut désigné par le type {moelle} a une forme désignée par une valeur de forme {allongée}.*

*Le défaut désigné par le type {moelle} a une position désignée par une valeur de position {centre de planche}.*

*Le défaut désigné par le type {moelle} a une orientation désignée par une valeur d'orientation {sens de fibre du bois}.*

*Le défaut désigné par le type {moelle} a une couleur désignée par une valeur de couleur {brune}.*

La figure 2 représente alors une formalisation partielle en NIAM/ORM de cet énoncé en phrases élémentaires.

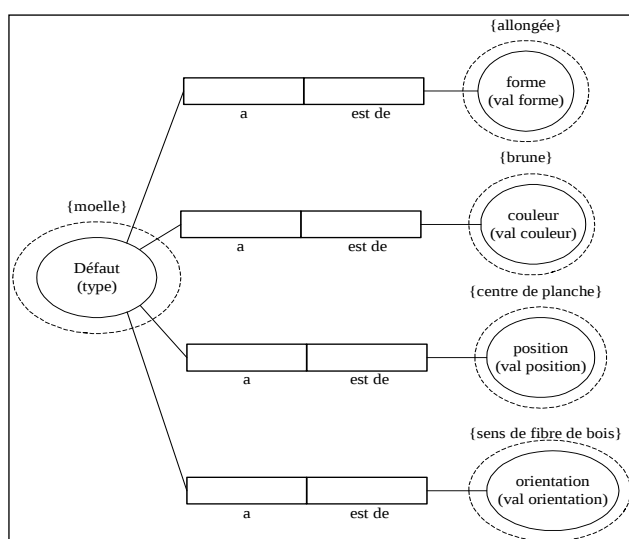


Figure 2 : modélisation partielle NIAM/ORM du défaut de type moelle

Nous faisons de même pour un autre défaut dont la description par l'expert est la suivante :

« Un nœud noir est un défaut de forme plutôt ronde, qui peut être aussi ovale, généralement éloigné des bords de la planche. Sa couleur est noire par rapport à la couleur du bois naturel, mais peut ne pas être entièrement noire ».

En appliquant le même principe de décomposition en phrases élémentaires, nous obtenons alors :

*Le défaut désigné par le type {nœud noir} a une forme désignée par une valeur de forme {ronde, ovale}.*

*Le défaut désigné par le type {nœud noir} a une position désignée par une valeur de position {éloigné des bords de la planche}.*

*Le défaut désigné par le type {nœud noir} a une couleur désignée par une valeur de couleur {noire}.*

La figure 3 représente une formalisation partielle en NIAM/ORM de cet énoncé en phrases élémentaires.

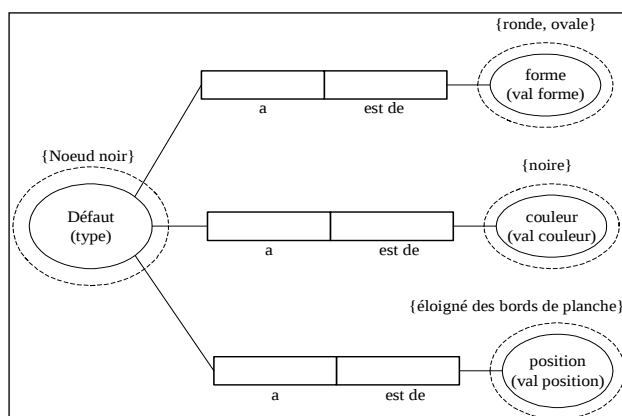


Figure 3 : modélisation partielle NIAM/ORM du défaut de type nœud noir

Nous pouvons remarquer que des invariants émergent de ces deux cas particuliers de défauts. Nous avons le nom du type de défaut, sa forme, sa couleur, sa position et son orientation. Bien que l'orientation ne soit pas commune à la définition des deux défauts, nous allons tout de même l'inclure au modèle générique en précisant qu'un défaut n'est pas obligatoirement caractérisé par une orientation. On peut même, à ce stade, se poser la question de savoir si cette notion d'orientation n'a pas de sens pour l'ensemble des défauts.

Nous allons donc créer un modèle générique issu de la généralisation de ces deux modèles particuliers.

Il faut maintenant introduire les notions de contraintes de totalité et d'unicité qui permettent de compléter l'information issue de la connaissance de l'expert. Pour ce faire, nous allons prendre l'exemple de la couleur.

Nous pouvons dire :

*Un défaut a une couleur, mais aussi qu'une couleur est de un défaut, ce qui a la même signification.*

Posons nous maintenant les questions sur l'existence de contraintes sur les deux formes de la phrase :

(•) Totalité : Chaque défaut a-t-il une couleur ? **Oui**  
 (↔) Unicité : Un défaut a-t-il une et une seule couleur ou une ou plusieurs ? **Une ou plusieurs**

(•) Totalité : Chaque couleur est-elle d'un défaut ? **Oui**  
 (↔) Unicité : Une couleur est-elle de un et un seul défaut ou un ou plusieurs ? **Un ou plusieurs**

Nous pouvons ainsi en déduire la formalisation NIAM de cette connaissance sur la couleur du défaut illustrée par la figure 4.

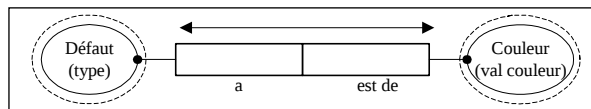


Figure 4 : modélisation NIAM/ORM du défaut et de sa couleur

En faisant de même pour les autres phrases, nous pouvons en déduire un modèle générique contenant toutes les informations issues de la connaissance de l'expert.

La figure 5 représente un extrait de la formalisation en NIAM/ORM de cette généralisation.

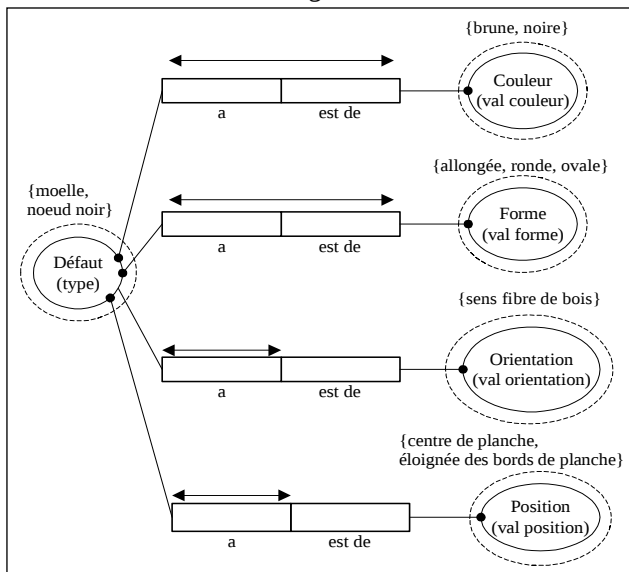


Figure 5 : extrait du modèle générique expert bois

Après avoir fait ces opérations pour chaque défaut énoncé par l'expert bois, nous pouvons obtenir une modélisation informationnelle de sa connaissance, généralisée à n'importe quel type de défaut, par la construction progressive du modèle conceptuel associé.

Il est à noter que ce modèle a été validé sur la base de son équivalent en Langage Naturel Binaire par l'expert ayant lui-même initialement exprimé sa connaissance en langage naturel.

## 2.2 Le modèle symbolique expert du procédé de vision

De même que pour le modèle symbolique de l'expert « métier bois », on recense les connaissances de l'expert du procédé de vision. Ces connaissances portent sur le système d'identification de défauts et, plus particulièrement, sur les paramètres utiles à la quantification des caractéristiques du défaut à reconnaître. Nous appliquons le même principe que pour le modèle symbolique de l'expert métier et nous aboutissons à une caractérisation de tous les défauts rencontrés mais, cette fois, du point de vue des paramètres du système de vision.

Nous avons interviewé les concepteurs du système de vision, et à partir de leur façon de caractériser chaque défaut, ou de leur connaissance sur les caractéristiques pertinentes pour identifier un défaut, nous avons construit un modèle général.

Voici ainsi un exemple qui permet de caractériser la couleur selon ce point de vue. L'expert procédé s'exprime de la manière suivante : « La couleur d'un défaut est mise en évidence par le paramètre renseignant sur le nombre de pixels noirs présents sur l'image du défaut. Plus le défaut est de couleur foncée, plus la valeur du paramètre le sera aussi par rapport aux valeurs obtenues pour les défauts plus clairs. »

Nous pouvons tirer les phrases élémentaires suivantes :

*Une couleur désignée par une valeur de couleur {foncée} se caractérise par un pourcentage de pixels noirs désigné par un nombre pourcentage de pixels noirs {important}. Un pourcentage de pixels noirs désigné par un nombre pourcentage de pixels noirs {important} caractérise une couleur désignée par une valeur de couleur {foncée}.*

La figure 6 représente une formalisation partielle en NIAM/ORM de cet énoncé.

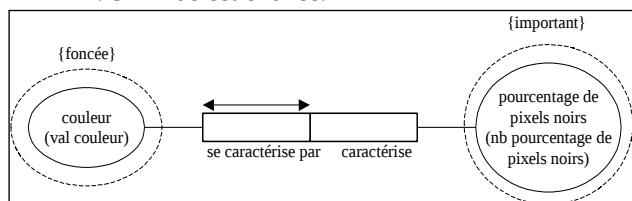


Figure 6: modélisation partielle NIAM/ORM expert procédé

Une fois tous les défauts caractérisés, en prenant en compte les contraintes de totalité et d'unicité, nous pouvons construire le modèle conceptuel associé permettant une généralisation de la connaissance de l'expert procédé. L'objectif de ce modèle est de relier les caractéristiques de vision aux règles floues.

De même que pour le modèle symbolique de l'expert métier, nous ne présentons pas le modèle dans sa totalité. Néanmoins, nous pouvons donner l'exemple de la modélisation du paramètre de forme qui

comporte, à lui seul, plus d'une dizaine d'idées et plus d'une dizaine de ponts de dénomination.

### 3 Le modèle numérique

Parallèlement aux connaissances exprimées par les modèles symboliques qui sont imprécises et incertaines car n'étant qu'une généralisation qualitative de la connaissance, nous disposons de mesures délivrées par la phase de segmentation qui sont entachées d'imprécision.

Afin de pouvoir faire un lien entre quantitatif et qualitatif, nous utilisons une méthode s'appuyant sur la théorie des sous-ensembles flous [10] permettant de supporter des raisonnements qualitatifs [18]. Ce processus est essentiel pour comparer les informations expertes (en langage naturel) avec la réponse du système (données floues). La figure 7 illustre le principe du modèle numérique.

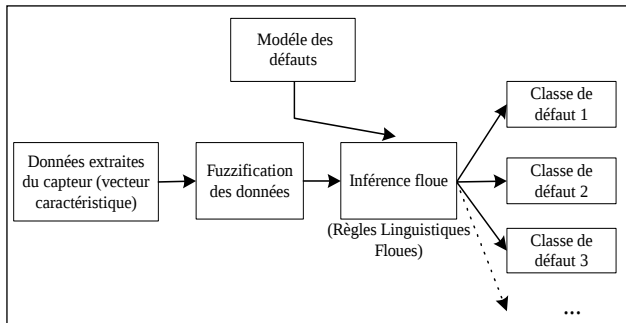


Figure 7: schéma explicatif du modèle numérique

#### 3.1 La méthode de Nozaki-Tanaka

Comme nous l'avons vu précédemment, nous utilisons un module de reconnaissance basé sur la logique floue. Notre choix s'est porté sur la méthode de Nozaki-Tanaka [13] [14]. Cette méthode de reconnaissance est une méthode itérative, mettant en oeuvre un mécanisme d'apprentissage supervisé. Elle permet de générer des règles floues du type « Si... alors... » définissant les défauts. Par exemple : « SI surface (petite) ET SI couleur (foncée) ALORS défaut (nœud noir) ».

Classiquement, lorsqu'on utilise un mécanisme de raisonnement flou de type Mandani ou Larsen [5], les règles sont généralement données par un expert, alors que la méthode de Nozaki-Tanaka permet d'obtenir automatiquement ces règles par apprentissage.

La méthode de Nozaki-Tanaka se déroule en trois étapes : la fuzzification des entrées, la génération des règles floues et l'ajustement. Le jeu de règles obtenu est ensuite utilisé pour effectuer la reconnaissance.

##### 3.1.1 Fuzzification des entrées

L'étape de fuzzification caractérise les variables linguistiques utilisées. Une variable linguistique [4] est défini par un triplet (V, X, Tv) où :

- V est la variable (surface, couleur, ...)
- X le référentiel (domaine de variation de V)
- Tv le vocabulaire choisi pour décrire de manière symbolique les valeurs de V (petit, grand, foncé, clair, ...)

Chaque terme T est un sous-ensemble flou défini sur X et caractérisé par sa fonction d'appartenance. Cette étape définit le nombre de décompositions de la variable considérée. Les termes sont choisis en rapport avec le vocabulaire utilisé par l'expert. L'intensité peut prendre le qualificatif « sombre » ou « claire », la surface peut être « petite », « moyenne » ou « grande ». Le choix du nombre de termes pour qualifier une variable est généralement empirique. On utilise une équirépartition des termes et un nombre de termes impair, relativement faible (trois ou cinq) pour ne pas avoir un nombre de règles trop important.

Dans notre cas, la décomposition est guidée par l'expertise qui définit le nombre de tendances (termes) pour une caractéristique (variable). Par exemple, il y a trois types de nœud « petit, intermédiaire et grand » qui conduisent à une décomposition des paramètres de vision associés à la taille en trois termes (figure 8). La position des fonctions d'appartenance est effectuée à partir d'une analyse du lot d'apprentissage. L'adaptation des termes aux données permet, sinon d'améliorer la reconnaissance, du moins de diminuer le nombre d'itérations de la méthode [2].

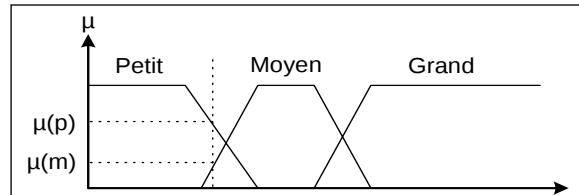


Figure 8 : fuzzification d'une entrée en 3 sous-ensembles

##### 3.1.2 Génération des règles floues

Cette deuxième phase permet la génération des règles de type « Si ... alors ». Avec ce système de règles floues, l'algorithme de Nozaki-Tanaka décrit la perception que le système a des défauts qui lui sont présentés. En effet, un lot de défauts identifiés par un expert est pris comme paramètre par l'algorithme de génération de règles floues, c'est le lot d'apprentissage.

Dans le cas où nous avons deux paramètres en entrée et une donnée de sortie, la forme générale de la règle floue est :

**SI** x1 est Ai **ET** SI x2 est Aj **ALORS** x3 est dans la classe de défaut Ci

Avec :

x1 et x2 les données d'entrées (par exemple la couleur et la surface)

x3 la donnée de sortie (nom du défaut)

Ai et Aj les sous-ensembles flous (par exemple 'petit' et 'grand')  
Ci la classe du i ème défaut

Chaque règle délivre une conclusion partielle qui est ensuite agrégée aux autres pour fournir la conclusion finale par un opérateur flou de disjonction. Dans notre cas nous avons choisi le max. L'inférence repose sur le modèle de Larsen [4] qui utilise comme opérateur de pseudo implication floue la composition max-produit.

### 3.1.3 Ajustement

L'ajustement représente la partie itérative de l'algorithme. Il consiste à tester, à partir du même lot de données, chaque défaut. Un mécanisme permet au système d'ajuster le découpage de l'espace de représentation en fonction des résultats obtenus [14].

Il est à noter que la méthode de Nozaki-Tanaka propose une phase supplémentaire d'affinage qui vise à augmenter le degré d'appartenance à la classe d'appartenance maximale en modifiant les pentes de la fonction d'appartenance. A l'extrême, cette phase permet d'avoir une représentation binaire. Nous n'utilisons pas cette étape car nous voulons conserver la gradualité des réponses. En effet, nous préférons ne pas classer un défaut dont le degré d'appartenance à la classe maximale est faible, plutôt que de faire une mauvaise identification.

Dans ce sens, nous nous rapprochons de la notion de rejet en ambiguïté [5] d'un classificateur bayésien. Néanmoins, l'idée est de lever ultérieurement cette ambiguïté par l'utilisation d'un module de niveau supérieur, exploitant, par exemple, une vue plus générale de la planche (mise en correspondance des faces, proximité des défauts du même type ...).

## 3.2 Inférence unique

Dans cette partie, les résultats sont obtenus à partir d'un moteur d'inférence comportant onze paramètres en entrée et quatorze classes de défauts en sortie. Sachant qu'en moyenne chaque paramètre est décomposé en trois termes, nous aboutissons à  $3^{11}$  règles linguistiques floues générées. Bien que les résultats obtenus soient très satisfaisants (voir paragraphe 4.2), cette méthode a l'inconvénient majeur de ne pas respecter la contrainte de temps industrielle fixée à 2 ms pour l'identification d'un défaut. En effet, nous obtenons environ 6 ms.

Un deuxième inconvénient rédhibitoire à nos yeux réside dans l'impossibilité d'interpréter l'ensemble des règles obtenues et ainsi de pouvoir les valider en les confrontant aux modèles symboliques.

C'est pourquoi nous proposons d'enchaîner hiérarchiquement plusieurs moteurs d'inférence

plus simples afin de constituer une arborescence réduisant ainsi le nombre de règles.

## 4 Méthodologie de construction d'une arborescence décisionnelle structurée par intégration de connaissances expertes

Cette méthodologie repose sur une méthode descendante, illustrée par la figure 9, qui part des connaissances les plus complexes pour aboutir à la construction du modèle numérique. Par connaissances complexes, nous considérons les connaissances porteuses d'informations ambiguës, en l'occurrence, les connaissances de l'expert métier du bois et de l'expert procédé de vision, exprimées en langage naturel. Grâce à la méthode NIAM, nous avons pu lever toutes ces ambiguïtés par la construction de modèles symboliques associés aux connaissances des différents experts.

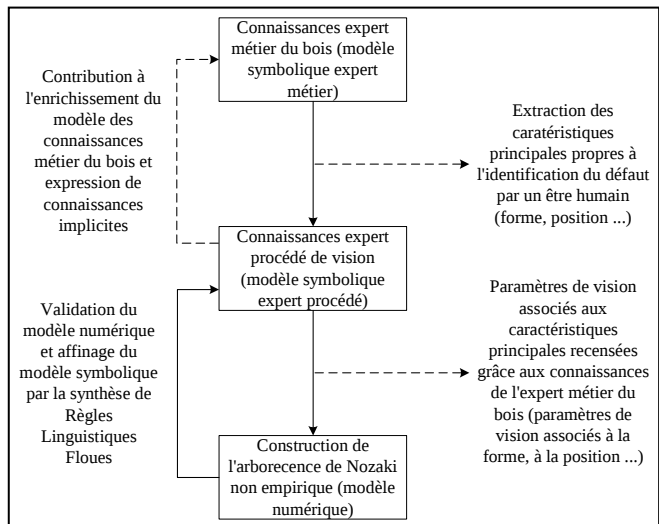


Figure 9 : schéma explicatif de la méthode descendante

### 4.1 Exemple d'intégration de connaissances expertes

Cet exemple a pour but de présenter notre méthode afin de différencier les défauts de type « noeud noir » (Black\_Knot), « petit noeud noir » (Small\_Black\_Knot) et « noeud noir en bord » (Edge\_Black\_Knot).

Dans une interview l'expert métier du bois a défini ces défauts de la manière suivante :

« Un noeud noir est un défaut de forme plutôt ronde, mais il peut être légèrement ovale. Sa couleur est noire par rapport à la couleur du bois naturel, mais ne peut être entièrement noire. Un petit noeud noir possède les mêmes caractéristiques que le noeud noir mais il a une petite taille. Un noeud noir au bord possède les mêmes caractéristiques que le noeud noir mais il se situe en bord de planche. »

A partir de ces phrases exprimées en langage naturel, nous pouvons construire un modèle symbolique

représentant les connaissances de l'expert du bois, illustré par la figure 10.

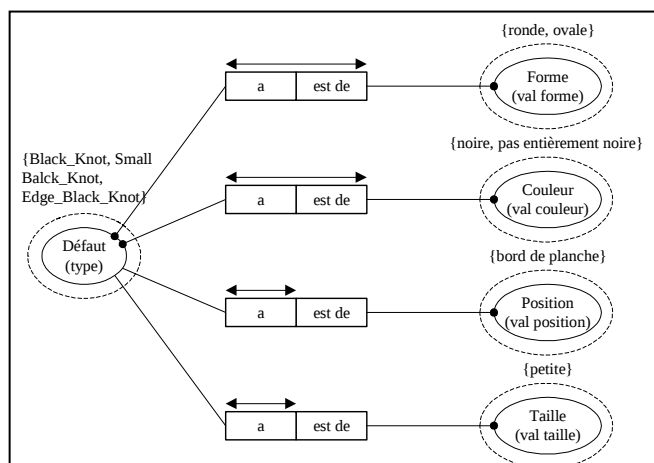


Figure 10 : modélisation NIAM/ORM des connaissances exprimées par l'expert du bois

D'après cette modélisation, nous pouvons remarquer que la famille « noeud noir » est décrite par quatre caractéristiques qui sont la forme, la couleur, la position et la taille. De plus, cette famille de nœuds est décrite de manière invariante par les caractéristiques de forme « ronde ou ovale » et de couleur « noire ou pas entièrement noire » (contraintes de totalité apposées sur l'objet « Défaut »). Seules les caractéristiques de position « bord de planche » et de taille « petite » permettent de différencier dans cette famille « nœuds noirs » les défauts de type « petit noeud noir » et « noeud noir au bord ».

Grâce aux connaissances de l'expert du procédé de vision, nous pouvons connaître les paramètres de vision qui caractérisent la taille et la position. Pour la taille, ces paramètres sont *Larg\_re*, *Long\_re*, *Pt\_axe* et *Surf*. Pour la position, seul le paramètre *C4* est utilisé.

Le noeud visant à identifier ces trois défauts est représenté par la figure 11.

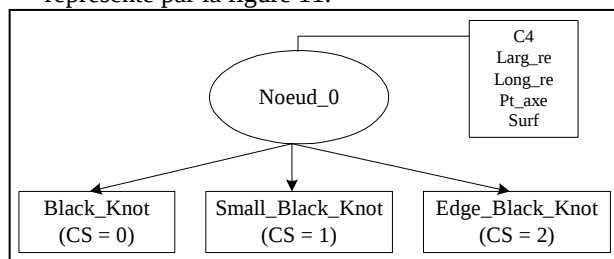


Figure 11 : schéma explicatif du noeud décisionnel visant à identifier les trois défauts

Les résultats de l'identification obtenus sont résumés dans les tableaux 1 et 2 qui suivent.

Tableau 1 : résultats de l'identification sur le lot d'apprentissage

| Nombre d'échantillons                                  | Pourcentage de bien classé | Nom du défaut    |
|--------------------------------------------------------|----------------------------|------------------|
| 10                                                     | 100.0                      | Black_Knot       |
| 4                                                      | 100.0                      | Small_Black_Knot |
| 10                                                     | 90.0                       | Edge_Black_Knot  |
| Pourcentage global d'identification : 95.9 (23 sur 24) |                            |                  |

Tableau 2 : résultats de l'identification sur le lot de généralisation

| Nombre d'échantillons                                  | Pourcentage de bien classé | Nom du défaut    |
|--------------------------------------------------------|----------------------------|------------------|
| 4                                                      | 100.0                      | Black_Knot       |
| 20                                                     | 100.0                      | Small_Black_Knot |
| 2                                                      | 50.0                       | Edge_Black_Knot  |
| Pourcentage global d'identification : 96.2 (25 sur 26) |                            |                  |

A ce niveau, nous pouvons nous demander si le modèle numérique construit est optimal (boucle de validation du modèle numérique sur la figure 9). Pour ce faire, nous proposons d'exploiter la matrice générée lors de la phase d'apprentissage à partir des lots de données, afin d'interpréter le mécanisme d'identification. Cette matrice est la traduction numérique des règles linguistiques floues, elle est formée de la manière suivante :

- les colonnes représentent les paramètres en entrée du noeud,
- les lignes représentent les règles.

A chaque prémisse est associé un numéro qui peut prendre les valeurs 0 ou 1 ou 2 en fonction qu'il représente les qualificatifs « petit », « moyen » ou « grand ». A chaque règle floue est associée une classe de sortie (CS) de défaut et un coefficient de confiance (CF) que le système accorde à la règle.

Par exemple, dans la matrice générée pour le noeud illustré par la figure 11, nous obtenons 51 règles linguistiques. Ces 51 règles sont de la forme suivante, illustrée par le tableau 3.

Tableau 3 : fragment de la matrice générée par l'exemple de la figure 11

| Prémises |         |         |        |      | CS  | CF      |
|----------|---------|---------|--------|------|-----|---------|
| C4       | Larg_re | Long_re | Pt_axe | Surf |     |         |
| 0        | 0       | 0       | 0      | 1    | 2   | 1.00000 |
| 0        | 0       | 1       | 0      | 1    | 2   | 1.00000 |
| 1        | 0       | 0       | 0      | 1    | 1   | 1.00000 |
| 1        | 0       | 2       | 0      | 1    | 1   | 1.00000 |
| ...      | ...     | ...     | ...    | ...  | ... | ...     |

Dans cet exemple, nous pouvons remarquer que la variation du troisième paramètre (*Long\_re*) de la valeur « petit » (0) à « moyen » (1) n'influe aucunement sur la classe de sortie, de même pour une variation de « petit » à « grand » (2).

Sur 51 règles générées, 46 d'entre elles considèrent donc que le paramètre *Long\_re* influe peu sur l'identification de la classe de sortie, laissant ainsi supposer qu'il n'est pas discriminant.

Pour le vérifier, nous décidons alors de créer un nouveau noeud en supprimant le paramètre *Long\_re* de la liste des paramètres d'entrée. Nous obtenons des résultats identiques aux tableaux 1 et 2 indiquant

que ce paramètre est inutile et ainsi permet d'obtenir un modèle numérique optimal pour l'identification des trois défauts. De plus, nous pouvons désormais affiner la connaissance de l'expert procédé de vision en supprimant ce paramètre vision de la liste des paramètres caractérisant la taille d'un défaut.

## 4.2 Résultats de l'arborescence complète

Les essais réalisés montrent que le modèle numérique fournit des résultats comparables aux méthodes actuellement utilisées sur les systèmes existants. Il offre, en outre, une facilité de paramétrage avec la phase d'apprentissage simplifiant la phase d'installation.

Nous avons construit trois types de modèles numériques :

- une arborescence de référence, en terme de taux d'identification, à un noeud avec en entrée tous les paramètres de vision et en sortie toutes les classes de défauts. Il est à noter que cette arborescence ne convient pas en terme de temps de traitement.
- une arborescence empirique construite sans analyse des connaissances avec pour principal objectif le respect de la contrainte de temps
- une arborescence structurée avec intégration de connaissances métier prenant en compte la contrainte de temps industrielle

Pour des raisons de confidentialité, nous ne pouvons exposer l'arborescence complète permettant d'identifier les quatorze défauts, mais le tableau 4 résume les performances obtenues.

Tableau 4 : résultats comparatifs des pourcentages globaux d'identification des trois arborescences

|                                | Méthode point (*) | Un noeud | Empirique | Structurée |
|--------------------------------|-------------------|----------|-----------|------------|
| Apprentissage                  | 72 %              | 99.1 %   | 94.4 %    | 93.5 %     |
| Apprentissage + compatibilité  |                   | 100 %    | 94.4 %    | 98.1 %     |
| Généralisation                 |                   | 68.8 %   | 63.6 %    | 66.5 %     |
| Généralisation + compatibilité |                   | 91.5 %   | 90.9 %    | 93.2 %     |

(\*) : la méthode point est une méthode non itérative dont le taux global de reconnaissance est obtenu à partir d'un seul lot de données.

Ce tableau contient une information supplémentaire appelée compatibilité qui prend en compte les compatibilités entre les défauts (par exemple un noeud cerné peut être considéré comme similaire à un petit noeud noir). Cette information est fournie par les experts en réponse à une confusion d'identification acceptable par les clients.

Nous obtenons les meilleurs résultats en généralisation pour une arborescence avec intégration des connaissances métier et une fuzzification adaptée aux variables caractéristiques puisque le taux de reconnaissance global atteint

alors 93%. De plus, en diminuant le nombre total de règles, nous avons abaissé le temps moyen d'identification à 1,3 ms par défaut, respectant ainsi la contrainte de temps industrielle.

Ces essais ont été menés sur un lot de 107 défauts pour l'apprentissage et un lot de 176 défauts pour la généralisation, mis à notre disposition par la société Luxscan Technologies.

## 4.3 Méthode du « laissé pour compte »

Au vu du nombre relativement faible d'échantillons, nous utilisons la méthode du « laissé pour compte » [5] pour calculer le taux de reconnaissance global. Ce taux représente l'association du taux de mémorisation et du taux de généralisation.

Pour ce faire, nous utilisons un lot unique de 283 défauts regroupant les 107 défauts du lot d'apprentissage et les 176 défauts du lot de généralisation. Les résultats obtenus sont résumés dans le tableau 5.

Tableau 5 : résultats des pourcentages d'identification avec la méthode du « laissé pour compte »

|                       | Sans compatibilité des défauts | Avec compatibilité des défauts |
|-----------------------|--------------------------------|--------------------------------|
| Taux d'identification | 84.1 %                         | 95.1 %                         |

Ces résultats montrent des taux d'identification sensiblement identiques à ceux obtenus à partir des deux lots distincts utilisés dans le paragraphe 4.2 indiquant que les lots utilisées sont représentatifs des données.

## 5 Conclusion et perspectives

Nous avons proposé une méthodologie mettant en relation un modèle numérique obtenu à partir de données et un modèle symbolique issu de connaissances métier. Cette mise en relation s'effectue grâce à une méthode descendante destinée à l'intégration des connaissances métier pour la construction du modèle numérique basé sur la méthode de Nozaki.

Cette méthodologie montre un intérêt certain quant aux performances en terme de taux d'identification et en terme de performances de traitement de l'information. De plus, grâce à l'interprétation des règles linguistiques floues, nous pouvons connaître le fonctionnement du modèle numérique, le valider ou l'optimiser et affiner les connaissances métiers.

Nous avons vu également que la mise en relation du modèle numérique et du modèle symbolique n'est possible que par la construction d'un modèle intermédiaire modélisant les connaissances de l'expert vision, lien entre le métier et l'application industrielle.

Le modèle symbolique des connaissances métier peut être aussi un moyen de mettre en relation les différents industriels du domaine du bois. En effet, il pourrait permettre l'utilisation d'un vocabulaire commun et ainsi faciliter la circulation de la

connaissance [3]. Par exemple, le terme « nœud cerné » de l'industriel 1 ne désigne-t-il pas le même défaut que le terme « nœud non adhérent » de l'industriel 2 ?

Quant au modèle symbolique du procédé de vision, il permet de capitaliser l'expérience des concepteurs de ce système. La base de données générée permet de recenser les caractéristiques utilisées et de les relier aux défauts pour lesquels elles sont pertinentes. Ainsi, en fonction des capteurs d'entrée, des essences de bois et d'autres paramètres d'environnement, l'expert vision peut s'appuyer sur ces connaissances pour concevoir ou configurer un nouveau système.

La méthode NIAM actuellement utilisée pour la formalisation des connaissances expertes montre cependant ses limites dès lors qu'il faut exprimer des contraintes autres qu'ensemblistes telles que des contraintes dites « procédurales ». Entrent dans ce cas les contraintes floues qui ne peuvent pas être directement décrites en NIAM. Cette lacune de NIAM ne nous permet donc pas d'intégrer complètement les modèles ainsi créés. Pour ce faire, nous envisageons de reprendre les travaux de Zgorzelski sur une extension floue de NIAM (« Fuzzy NIAM ») [19] et de les appliquer à une modélisation floue des modèles symboliques expert métier et expert procédé vision.

Par ailleurs, ces travaux appliqués au domaine de l'identification de défauts par vision sur des planches de bois pourraient être généralisés à tout autre domaine d'application où il existe un cloisonnement entre une expertise métier (le qualitatif) et les informations issues de capteurs (le quantitatif).

## Bibliographie

- [1] C. Attiogbé, H. Habrias, A. Vailly, « Exercices de style », *rapport interne n°139 de l'Institut de Recherche Informatique de Nantes*, 1996.
- [2] S. Babusiaux, G. Lerch, O. Siaud, G. Simon, « Fuzzification de données », *rapport interne du CRAN*, Nancy 1, 2000.
- [3] J.C. Blaise, « Apport d'une modélisation de l'information normative à l'intégration des règles de sécurité des machines en conception », *Thèse d'université*, Université Henri Poincaré, NANCY 1, 2000.
- [4] B. Bouchon-Meunier, « La logique floue et ses applications », *Addison-Wesley*, 1995.
- [5] B. Dubuisson, « Diagnostic, intelligence artificielle et reconnaissance des formes », *édition Hermès*, 2001.
- [6] S.Y. Foo, "A rule-based machine vision for fire detection in aircraft dry bays and engine compartments", *Knowledge-Based Systems* 9, pages 531-540, 1996.
- [7] H. Habrias, « Le modèle relationnel binaire. Méthode I.A. (NIAM) », *éditions Eyrolles*, 301p., 1998.
- [8] T.A. Halpin, "Object Role Modeling (ORM/NIAM)", *Handbook on Architectures of Information Systems*, 1998.
- [9] T.A. Halpin, "Information Modelling and Relational Databases", published by *Morgan KAUFMAN edition*, 2001.
- [10] A. Kaufmann, « Introduction à la théorie des sous-ensembles flous », *édition Masson*, 1975.
- [11] H. Kauppinen, H. Rautio, O. Silvén, "Non-segmenting defect detection and SOM based classification for surface inspection using color vision", *Conference on Polarization and Color Techniques in Industrial Inspection (SPIE 3826)*, Munich, Germany, pages 270-280, 1999.
- [12] J. Lampinen, S. Smolander, M. Korhonen, "Wood surface inspection system base don generic visual features", *International Conference on Artificial Neural Networks ICANN'95*, Paris, 1995.
- [13] K. Nozaki, H. Ishibuchi and H. Tanaka, "Trainable Fuzzy Classification Systems based On Fuzzy If-Then Rules", *Proc. Of the 3<sup>rd</sup> IEEE Int. Conf. On Fuzzy Systems*, Orlando, USA, pages 498-502, 1994.
- [14] K. Nozaki, H. Ishibuchi and H. Tanaka, "A Simple but powerful heuristic method for generating fuzzy rules from numerical data". *Fuzzy sets and systems* 86, pages 251-270, 1997.
- [15] D.T. Pham, S. Sagiroglu, "Training multilayered perceptrons for pattern recognition: a comparative study of four training algorithms", *International Journal of Machine Tools & Manufacture* 41, pages 419-430, 2001.
- [16] C.S. Wang, X.K. Gong, "A fuzzy approximate reasoning model for a rule-based system in laser threat recognition", *Fuzzy Sets and Systems* 96, pages 139-146, 1998.
- [17] Z. Wen, Y. Tao, "Building a rule-based machine-vision system for defect inspection on apple sorting and packing lines", *Expert Systems with Applications* 16, pages 307-313, 1999.
- [18] L.A. Zadeh, "The concept of linguistic variable and its application to approximate reasoning" *Information sciences*, Vol. 8, p199-249, 1975.
- [19] M. Zgorzelski, Z. Zalewski "Fuzzy NIAM for real world data modeling", *I.S.A.T.A. 29<sup>th</sup> International Symposium on Automotive Technology and Automation*, Florence 1996.