

CONNAISSANCES ET DONNEES EXOGENES DANS L'INTERPRETATION D'IMAGES SATELLITE: LE SYSTEME EXPERT I. C. A. R. E.

DESACHY J. ¹

L'analyse par un photointerprète d'une image satellite, se fait en utilisant des informations extérieures à l'image (caractéristiques particulières de la zone étudiée, relief, géologie, facteurs d'occupation humaine, etc...) et ses connaissances sur les objets recherchés dans l'image (contexte favorable à la présence d'un objet en fonction des informations exogènes: relief, ensoleillement, proximité de voies de communication, etc,,).

Le but du système I.C.A.R.E. est justement de rajouter aux processus informatiques classiques d'interprétation d'images de télédétection, dont on connaît les résultats imparfaits, cette méthodologie de photointerprétation en permettant l'intégration de données exogènes et d'une méthode de raisonnement s'appuyant sur la connaissance des caractéristiques des objets recherchés, Dans une première phase du système, les "objets" recherchés sont des classes,

- Les informations exogènes peuvent consister en M.N.T., carte des sols, des routes, des zones urbaines. . . , sous forme numérisée (par exemple dans un S.I. G.) . Certaines connaissances seront calculées sur ces données (par exemple calcul du réseau hydrographique et débit) ou sur l'image (détection des zones urbaines) .
- La connaissance de l'expert sur les classes recherchées sera représentée sous forme de règles de production. La notion de connaissance imprécise étant prise en considération par le système.

Le principe de fonctionnement d'I.C.A.R.E. en "classifieur intelligent" suit les étapes suivantes:

- 1) Réalisation d'une classification supervisée classique mais pour tout point on conserve son degré d'appartenance $CFY(x,y)$ à chaque classe i possible,
- 2) On calcule le contexte de chaque pixel en relation avec les données exogènes et en utilisant les connaissances sur chaque classe i , on modifie ces $CFi(x,y)$ (celui ci est ainsi augmenté si le contexte du pixel est favorable à l'existence de la classe i),
- 3) Quand toutes les règles ont été activées pour tous les pixels et toutes les classes, on peut alors classer chaque pixel,

Le système peut aussi être utilisé en "résolution de problèmes", si la solution du problème peut s'exprimer sous forme d'image de priorités et que les connaissances utilisées sont liées à des données géocodées ou des informations extraites de l'image.

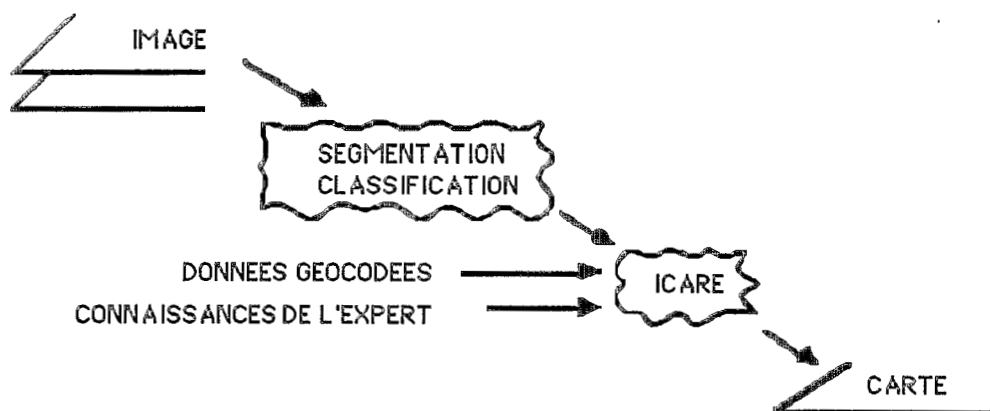
¹Laboratoire I.R.I.T. Université P.Sabatier 118, route de Narbonne 31062 TOULOUSE-cédex tél. 61 55 65 99

Le système I.C.A.R.E. (Image Cartographie Expert) est un système d'interprétation d'images satellite. Son but est l'aide à la cartographie automatique thématique. Son domaine d'application privilégié est en fait la classification thématique des images (par exemple produire une carte de végétation).

Il peut être conçu comme un système qui "corrige" les erreurs dans les méthodes classiques de classification ou de segmentation d'images, dont on sait qu'elles donnent des résultats peu satisfaisants.

Une retombée intéressante de ce système est sa capacité à résoudre des problèmes où les données utilisées pour le raisonnement se trouvent dans un système d'informations géographiques et dont la solution apparaît sous forme d'une carte de "priorités".

On peut intégrer des données exogènes dans le processus de classification lui-même, mais dans ce cas il est délicat d'intégrer des connaissances [CAROPP086], ou bien, procéder en deux étapes: une première étape classique de classification (supervisée ou non) suivie d'une étape qui va consister à améliorer cette classification. C'est cette deuxième forme que nous avons choisi car elle seule, permet d'intégrer les connaissances et le savoir-faire de l'expert dans l'interprétation de l'image. La représentation la plus schématique du système dans son aspect "cartographie thématique", peut être donnée par la figure suivante:



Nous allons présenter un premier système que nous avons réalisé sur un exemple concret de cartographie à partir d'une image satellite. Ce premier système a permis de vérifier tout l'intérêt de l'approche décrite au niveau de l'amélioration des résultats, mais il avait le défaut majeur de ne pas être adaptable à une autre région que la région particulière étudiée et à d'autres connaissances que celles utilisées dans ce cas précis, sans avoir à réécrire la plus grande partie du logiciel [FISSE84(1)][FISSE84(2)][DESACHY86].

1-PREMIERS PAS VERS LE SYSTEME ICARE

Dans ce premier logiciel, il s'agissait de réaliser une carte de végétation d'une zone subtropicale de l'Inde, la région de Palni, et ceci à l'aide d'images satellite LANDSAT MSS. Quatorze classes de végétation étaient définies et pour chacune de ces classes étaient fournis des échantillons. Pour la même zone on disposait de données bibliographiques sur les caractéristiques de végétation, sous forme de descriptions de zones écofloristiques décrivant un milieu d'un type donné et les différentes végétations possibles.

On disposait d'autre part de cartes relativement anciennes (colonisation anglaise), qui nous ont permis de construire un modèle numérique de terrain à partir des courbes de niveau.

Nous allons maintenant décrire rapidement comment nous avons procédé dans ce premier système intégrant des connaissances et des informations hors-image pour procéder à une classification thématique:

La base de connaissances (constituée des descriptions des caractéristiques de chacune des végétations recherchées) a été construite à partir d'un ensemble de descriptions de zones écofloristiques.

La base de données (constituée de modèles numériques, c'est à dire de données de type "image") a été construite à partir de cartes anciennes de la région. On a pu ainsi construire le modèle numérique de terrain (par extraction de points sur courbes de niveau, puis interpolation), à partir de là on a pu extraire les crêtes et les vallées... et d'autre part le modèle numérique des zones écofloristiques.

Cette base de connaissances et cette base de données vont intervenir en trois points de notre méthode:

D'abord au niveau de la classification supervisée initiale, ou va intervenir le modèle numérique des zones écofloristiques. En effet pour chaque point de l'image, on va faire intervenir son "numéro" de zone écofloristique comme un paramètre supplémentaire. A ce niveau du traitement, nous appliquons une méthode de classification supervisée classique (nous l'appellerons préclassification) où chaque pixel est décrit par ses radiométries, sa texture et la zone écofloristique à laquelle il appartient. Pour chaque pixel nous ne conservons que les trois classes pour lesquelles les probabilités d'appartenance sont les plus élevées.

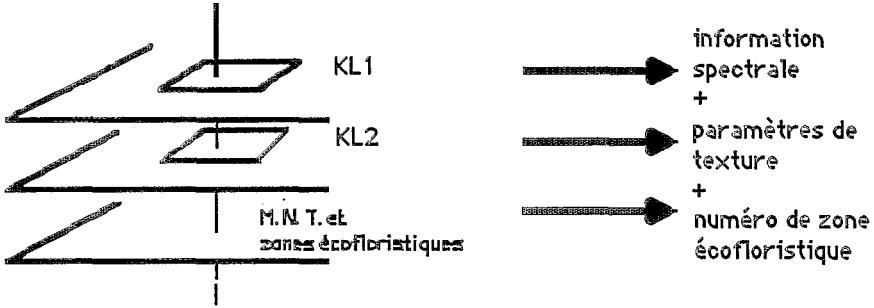
Puis on utilise les bases de données et de connaissances pour confirmer ou infirmer la préclassification, en utilisant les caractéristiques de localisation de chaque classe de végétation ou de culture. On utilise ici des caractéristiques dites de niveau 1, qui ne concernent que la localisation du pixel lui-même.

Puis on confirme ou infirme la classification fournie par l'étape précédente, en utilisant des caractéristiques dites de niveau 2 faisant intervenir des notions de surface ou de forme (par exemple "grandes étendues" ou bien "reliques" pour une classe de végétation), et qui nécessitent d'avoir déjà réalisé une classification.

Nous allons maintenant décrire les différentes étapes du processus:

1.1 - DEFINITION DES CLASSES

Chaque pixel est caractérisé par ses radiométries (après une transformation de KARHUNEN-LOEVE où on ne garde que les deux premiers canaux transformés), la texture de son voisinage et enfin le numéro de zone écofloristique où il se trouve.

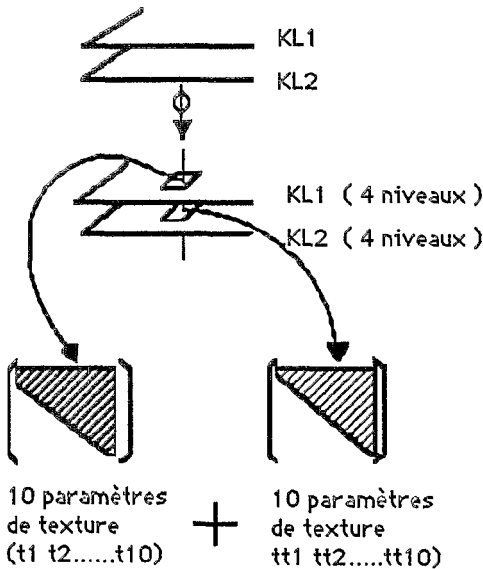


Les paramètres de texture sont extraits des matrices de cooccurrences de niveaux de gris de Harralick [HARRALICK79] [DESACHY80] [DESACHY82] .

Les matrices de cooccurrences que nous avons utilisé, étaient définies par une relation spatiale de type "distance 1"; l'élément (i,j) d'une matrice d'Harralick étant défini par:

$$M(i,j) = \text{card} [((x,y), (x',y'))], \quad \text{vérifiant} \quad \begin{aligned} d((x,y), (x',y')) &= 1 / \\ (x,y) &= i-1 \text{ et } l(x',y') = j-1 \end{aligned}$$

Les paramètres de texture sont définis par les éléments des matrices de cooccurrences après que les transformées de K.L. ont été réduites à quatre niveaux de gris [DESACHY80].



Chaque pixel est alors caractérisé par le vecteur:

$$\begin{array}{ccc} (I_{KL1}(x,y) \ t1 \ t2 \dots t10 & I_{KL2}(x,y) & t1 \ t2 \dots t10 \ z) \\ \leftarrow \text{KL1} \rightarrow & & \leftarrow \text{KL2} \rightarrow \end{array}$$

Z: numéro de zone écofloristique

chaque classe est définie par n pixels échantillons

$$(I_{KL1}^1 \ t1 \dots t10 \ I_{KL2}^1 \ t1 \dots t10 \ Z_1)$$

• • • • •

• • • • •

$$(I_{KL1}^n \ t1 \dots t10 \ I_{KL2}^n \ t1 \dots t10 \ Z_n)$$

Remarquons à ce niveau que si un pixel échantillon se trouve à proximité de la frontière de deux zones écofloristiques, Z_a et Z_b , on considérera que la zone écofloristique pour ce pixel peut être indifféremment Z_a ou Z_b , ce qui permet de définir des frontières non rigides entre zones écofloristiques.

Une classe va donc être définie par:

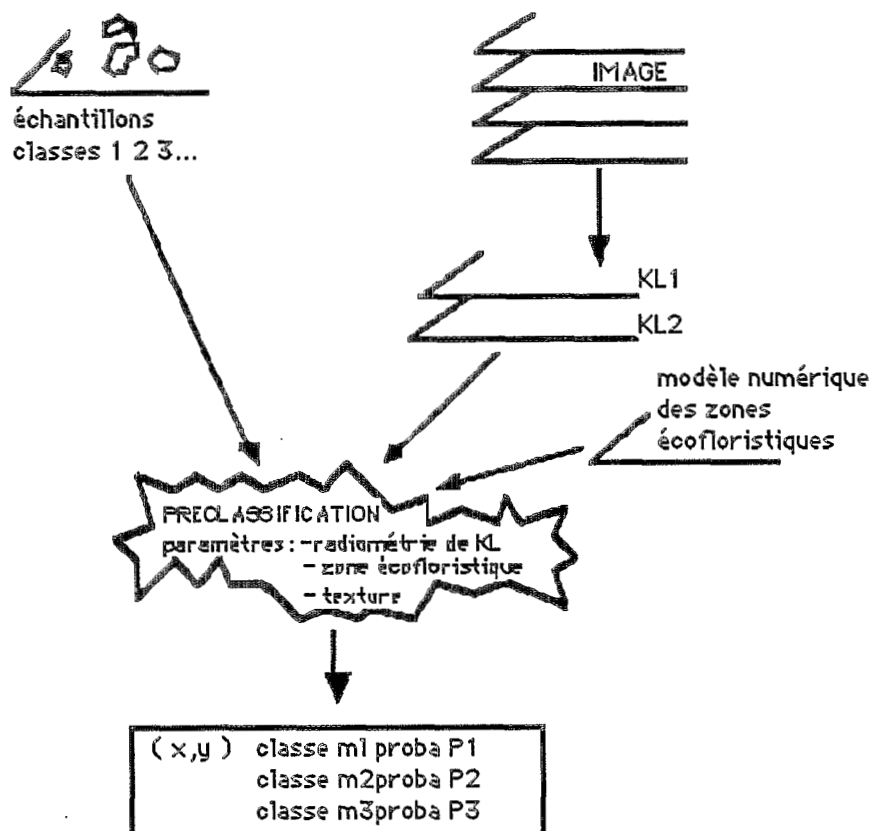
$$\left[\begin{array}{c} \left(\begin{array}{c} \mu_{I_{KL1}} \\ \mu_1 \\ \vdots \\ \mu_{10} \end{array} \right) \quad \left(\begin{array}{c} \sigma_{I_{KL1}}^2 \\ \sigma_1^2 \\ \vdots \\ \sigma_{10}^2 \end{array} \right) \\ \left(\begin{array}{c} \mu_{I_{KL2}} \\ \mu \mu_1 \\ \vdots \\ \mu \mu_{10} \end{array} \right) \quad \left(\begin{array}{c} \sigma_{I_{KL2}}^2 \\ \sigma \sigma_1^2 \\ \vdots \\ \sigma \sigma_{10}^2 \end{array} \right) \\ L = \{Z_1 \dots Z_m\} \end{array} \right]$$

L est l'ensemble des numéros de zones écofloristiques où on a trouvé des pixels échantillons.

μ et σ^2 représentant moyenne et variance des paramètres correspondants.

1.2 - PRECLASSIFICATION

Elle est résumée dans le schéma suivant:



La méthode de classification utilisée est une méthode géométrique. Pour chaque pixel on mesure la distance entre la signature de ce pixel et chaque classe (qui doit avoir au moins un pixel échantillon dans la meme zone écofloristique) par la formule suivante:

$$d(I(x,y), \text{classe}) = \gamma_1 \left(\frac{(I_{KL1}(x,y) - \mu_{I_{KL1}})^2}{\sigma_{I_{KL1}}^2} + \frac{(I_{KL2}(x,y) - \mu_{I_{KL2}})^2}{\sigma_{I_{KL2}}^2} \right) + \gamma_2 \left(\sum_{j=1}^{10} \frac{(t_j - \mu_j)^2}{\sigma_j^2} + \sum_{j=1}^{10} \frac{(tt_j - \mu_{tt_j})^2}{\sigma_{tt_j}^2} \right)$$

γ_1 et γ_2 étant deux facteurs permettant de moduler les contributions relatives des radiométries et des paramètres de texture. Si on suppose que le pixel (x,y) se trouve dans une zone écofloristique de n° Z(x,y), alors à la fin de la préclassification, à chaque pixel est associé un ensemble n de classes possibles

auxquelles il peut être affecté (éventuellement on ne garde que les trois classes pour lesquelles la distance est la plus petite).

$\Omega_n = \{ \text{classes } m_j / m_j \text{ est une classe se trouvant dans la zone écofloristique } Z(x,y) \text{ ET } d(I(x,y), \text{classe } m_j) \leq S \}$

Le seuil S (c'est à dire la distance au delà de laquelle le pixel (x,y) traité n'appartiendra pas à la classe m_j) est fixé dans l'hypothèse de "classes gaussiennes" (on suppose donc que chaque classe définie suit une loi de probabilités de Gauss à 22 dimensions)

On suppose donc que pour chaque pixel (x,y) on arrive à la "préclassification" suivante:

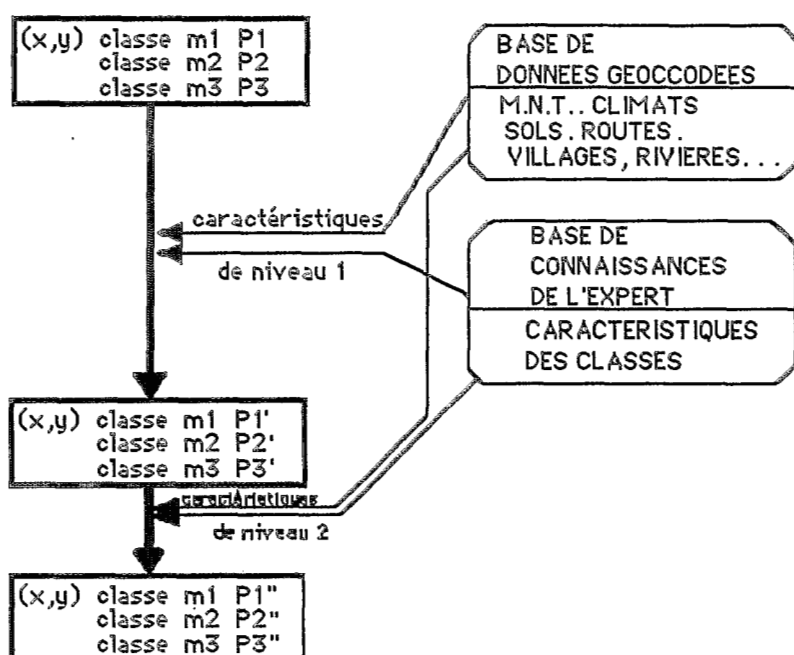
$$\left\{ \begin{array}{ll} \text{classe } m_1 & d_1 = d(I(x,y), \text{classe } m_1) \leq S \\ \text{classe } m_2 & d_2 = d(I(x,y), \text{classe } m_2) \leq S \\ \text{classe } m_3 & d_3 = d(I(x,y), \text{classe } m_3) \leq S \end{array} \right\}$$

et pour avoir des probabilités, on aura:

$$\left\{ \begin{array}{ll} \text{classe } m_1 & P_1 = 1 - (d_1/S) \\ \text{classe } m_2 & P_2 = 1 - (d_2/S) \\ \text{classe } m_3 & P_3 = 3 - (d_3/S) \end{array} \right\}$$

1.3 - CLASSIFICATION

C'est à cette étape du processus que nous allons faire intervenir les connaissances de l'expert sur les caractéristiques de chacune des classes recherchées. Un pixel classé dans une classe m_1 avec une probabilité P_1 dans la préclassification se verra attribuer une probabilité supérieure si sa localisation est favorable à la classe m_1 (inférieure sinon), de même pour les classes m_2 et m_3 . Ce traitement s'opérant en deux étapes, une pour chaque type de caractéristique (niveau 1 et 2) suivant le schéma suivant:



Nous allons décrire d'abord comment nous avons décidé de coder les connaissances de l'expert:

---> Construction de la base de connaissances:

Le problème est de formaliser pour chaque classe de la région d'étude, des expressions du type suivant décrivant les caractéristiques des classes de la région:

- classe 1: seulement hauts de crêtes ou fonds de vallées mais reliques et altitude supérieure à 1800m
- classe 2: seulement sur relief modéré et altitude entre 1000 et 1900m
- classe 3: seulement fonds de vallée et surtout versants est ou sud et altitude entre 900 et 1800m.
- classe 4: uniquement pentes escarpées et altitude de 900 à 1800m
- classes 5 et 6: uniquement près des agglomérations.
- classe 7: largement dominant sur le plateau et altitude > 1800m
- classe 8: surtout vallées des versants Nord. altitude de 900 à 1800m
- classe 9: uniquement pentes escarpées ou hauts de versants et altitude < 900m
- classe 10: Dépressions irriguées et altitude < 900m
- classe 11: Zones non irrigables et altitude < 900m
- classe 12: commun près des villages ou des routes et altitude < 900m
- classe 13: fréquents sur tous les versants et altitude de 900 à 1800 m
- classe 14: surtout versants nord et altitude < 900 m

Nous avons formalisé la représentation de ces expressions de caractéristiques des classes recherchées de façon à ce qu'elle satisfasse à deux critères:

- > permettre à l'expert d'exprimer relativement facilement sa connaissance sur les classes recherchées .
- > faciliter le calcul pour chaque point de la valeur de son contexte pour chaque classe possible.

Ces contraintes nous ont amené à définir quatre types de caractéristiques générales:

>> CTO:

Caractéristique de Type Objet, indiquant l'entité intervenant dans le contexte favorable (ou non) d'une classe, cette entité étant repérable dans la base de données géocodées dont on dispose.

exemples:

- Vallée
- Versant - ouest
- sud.....
- Route
- Agglomération

certaines de ces entités étant directement repérables sur une strate de la base de données ou bien calculable (par ex. un versant ...).

» CTR:

Caractéristique de Type Relationnel décrivant la relation entre la classe à laquelle on s'intéresse et l'objet défini par CTO. Chaque CTR inférant une technique de calcul de la réalisation du "contexte favorable correspondant".

exemples;

- Sommet
- Fonds de
- Prés de
- En face de
-

Une relation du type **prés de route** par exemple. entraînera un type de calcul de vérification de ce contexte faisant intervenir des notions de distance.

>> CTP:

Caractéristique de Type Présence. C'est en fait cette caractéristique qui nous permet de faire intervenir l'incertitude que l'on peut avoir sur la description des caractéristiques d'une classe.

exemples:

- Seulement
- Principalement
- Rarement
- Jamais
- Commun
- La plupart du temps
-

>> CTS:

Caractéristique de Type Spatial. Cette caractéristique permet d'exprimer certaines caractéristiques de classes, que nous qualifions de niveau 2 et qui font intervenir des notions de surface, de forme, etc ...

ex:emple: grandes étendues, reliques.....

A partir de ces quatre caractéristiques CTO,CTR,CTP,CTS, il va être possible de représenter les connaissances que l'on a sur chaque classe (connaissances que l'on a séparé en deux types

niveau 1 : concerne un point indépendamment des points voisins.

niveau 2 : les points voisins doivent être connus et donc classés, ces connaissances ne peuvent donc intervenir que si l'on a déjà fait une classification).

La représentation de la connaissance que l'on a sur une classe donnée prendra alors la forme suivante: **caractéristique globale de niveau 1:**

((CTP.CTR.CTO) et (CTP.CTR.CTO) et)

ou ((CTP.CTRCTO) et (CTP.CTR.CTO) et)

ou

En fait, une "union d'intersections de caractéristiques élémentaires" ayant la forme (CTP.CTR.CTO). par exemple classe 12,

"commun prés des villages et des routes" s'exprimera sous la forme:

(Commun.prés.village) et (commun.prés.route)

(CTP1.CTR1.CTO1) et (CTP2.CTR2.CTO2)

caractéristique globale de niveau 2:

(CTP.CTS) et (CTP.CTS) et

par exemple: : "reliques "

Reprenons un exemple faisant intervenir des connaissances de niveau 1 et 2 exemple classe 1,

"seulement hauts de crêtes ou fonds de vallées mais reliques" donnera

caractéristique de niveau 1:

(SEULEMENT. HAUT. CRETE) ou (SEULEMENT. FONDS .VALLEES)

caractéristique de niveau 2:

(UNIQUEMENT. RELIQUES)

Voyons maintenant comment on utilise cette formalisation de la connaissance pour réaliser une "amélioration" de la préclassification.

---> UTILISATION DE LA BASE DE CONNAISSANCES POUR LA CLASSIFICATION

Les caractéristiques CTP,CTR,CTO sont utilisées pour améliorer les résultats de la préclassification en modifiant les probabilités d'appartenance aux classes de chaque pixel.

Le principe étant que la probabilité correspondante sera augmentée si le contexte du pixel est favorable à une classe donnée et diminuée dans le cas contraire. A ce niveau du traitement nous ne prenons en compte pour chaque classe que les caractéristiques de niveau 1.

Rappelons que pour chaque pixel nous ne gardons que trois possibilités de classes (les plus favorables):

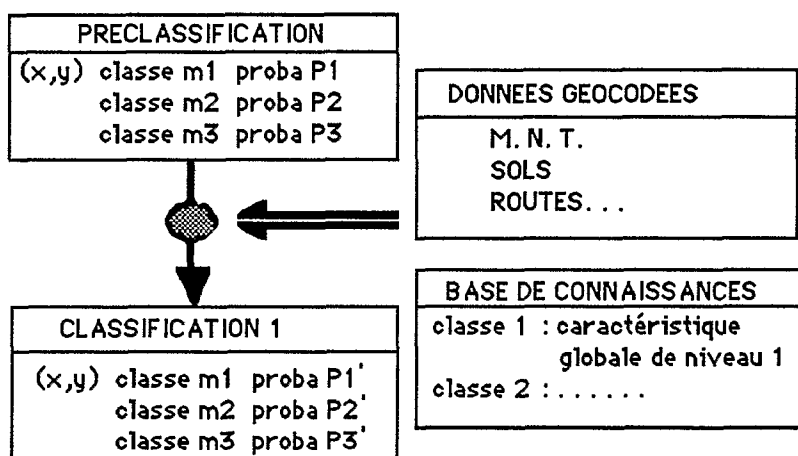
- classe m_1 probabilité P_1
- classe m_2 probabilité P_2
- classe m_3 probabilité p_3

Pour chaque classe m_i possible, on calcule une nouvelle probabilité d'appartenance à la classe m_i , P'_i à partir de P_i et en prenant en compte

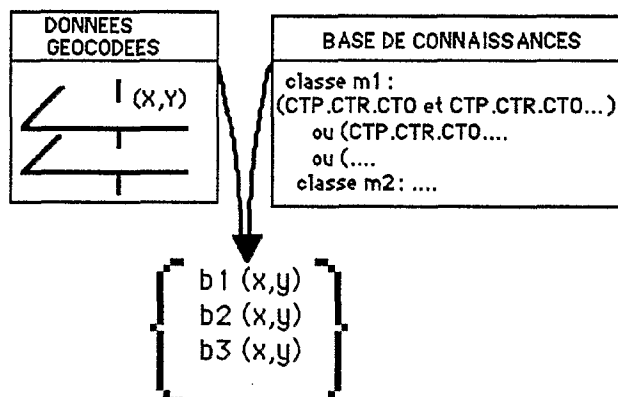
- la caractéristique globale de niveau 1 de la classe m_i
- la localisation du pixel en relation avec la base de données géocodées.

On cherche en fait un facteur correctif b_i tel que

$$P'_i = b_i * P_i$$



Voyons maintenant comment est réalisé le calcul du facteur correctif b_i



Le calcul va commencer au niveau de chaque caractéristique partielle de la forme (CTP_i, CTR_i, CTO_i) pour produire un facteur correctif partiel $pcari$.

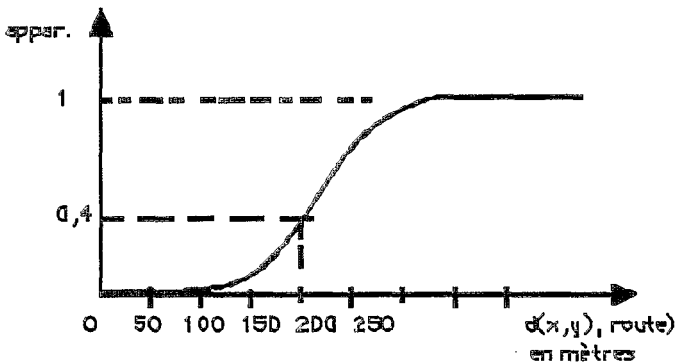
* CTP_i définit une "probabilité de présence" P_{pres} si la condition CTR_i, CTO_i est vraie à 100% et $(1-ppres)$ si elle est fausse à 100% . Ces valeurs de $ppres$ étaient définies par le tableau suivant:

CTP	seulement	principalement	prédominant	fréquent	présent	jamais
P_{pres}	1	0.85	0.7	0.55	0.4	0

* CTR_i induit un "niveau d'appartenance" à l'objet CTO_i , soit appar, qui est calculé à l'aide de la logique des sous-ensembles flous (d'autres techniques sont ajoutées dans le système ICARE)

exemple

"près de route"
 CTR_i, CTO_i



par exemple si le pixel (x,y) est à 150 mètres d'une route alors le degré de réalisation de la caractéristique "près de route", appar, est de 0.4. Alors le facteur correctif partiel $pcari$ pour la caractéristique partielle CTP_i, CTR_i, CTO_i est défini par:

$$pcari = ppres * appar + (1 - ppres) * (1 - appar)$$

Le facteur correctif total, bi , est alors calculé sur la base de tous les facteurs correctifs partiels $pcari$ (en prenant le minimum sur les facteurs correctifs partiels des caractéristiques partielles liées par un "et" et le maximum pour celles qui sont liées par "ou") suivant le schéma suivant:

$$\begin{array}{c}
 \text{(CTP1.CTR1.CTO1 } \Omega \text{ CTP2.CTR2.CTO2 } \Omega \text{ ...) } \cup \text{(...) } \cup \dots \\
 \hline
 b1' = \min_k (p_{car_k}) \qquad \qquad \qquad b2' \\
 \hline
 b = \max_j (b_j') * ech
 \end{array}$$

où ech est une constante qui permet de jouer sur les valeurs limites possibles de b, car la valeur $\max(b'_j)$ est toujours inférieure à 1. Donc dans le passage de la probabilité p_i (probabilité d'appartenance du pixel (x,y) à la classe m_j), la nouvelle probabilité P'_i serait toujours inférieure à P_i puisque $P'_i = b_j * p_i$, ce qui ne perturberait pas le résultat final puisqu'un pixel situé dans des conditions favorables verrait sa probabilité P_i rester identique alors qu'elle diminuerait dans le cas contraire. Mais sachant qu'un contexte indifférent pour une classe donnée fournira une valeur de b égale à 0.5 si ech est égal à 1, nous avons utilisé une valeur de ech égale à 2, ce qui permet d'avoir $P'_i > P_i$ quand le contexte est favorable et $P'_i < P_i$ quand il est défavorable.

A la fin de la classification utilisant les caractéristiques de niveau 1 on obtient la situation suivante pour le pixel (x,y):

$$\left\{ \begin{array}{lll}
 (x,y) & \text{classe } m1 & \text{probabilité } p'1 = b1 * p1 \\
 & \text{classe } m2 & \text{probabilité } p'2 = b2 * p2 \\
 & \text{classe } m3 & \text{probabilité } p'3 = b3 * p3
 \end{array} \right\}$$

On procède alors à la deuxième étape qui consiste à faire intervenir les caractéristiques de niveau 2 des classes recherchées. Dans notre exemple, il s'agissait uniquement de notions de surface.

De façon analogue au calcul du niveau 1, pour chaque classe possible m_i on calcule une nouvelle probabilité d'appartenance du pixel (x,y) à la classe m_i , P''_i à partir de P'_i

- en prenant en compte la caractéristique globale de niveau 2 de la classe m_i .
- la taille de la région homogène dans laquelle se trouve le pixel (x,y).

$$P''_i = k_i * P'_i$$

Une caractéristique globale de niveau 2 a la forme suivante:

$$(CTP_1.CTS_1) \text{ et } (CTP_2.CTS_2) \text{ et } \dots$$

Pour chaque caractéristique partielle on calcule un facteur correctif partiel:

$$p_{car_i} = p_{pres_i} * appar_i + (1 - p_{pres_i}) * (1 - appar_i)$$

ou $ppres_i$ est la "probabilité de présence" décrivant la confiance (CTP_i) qu'on a dans l'affirmation " CTS_i " (1: sur à 100%; 0: faux à 100%) et $appar_i$ est le degré de réalisation de CTS_i

par exemple,

"reliques" sera conçu comme "uniquement reliques" et donc $ppres = 1$.
Quant à $appar$, il sera calculé en fonction de la surface S de la région homogène où se trouve le point

SI $(S_1 \leq S \leq S_2)$

ALORS $appar = (S - S_1) / (S_2 - S_1)$

SINON $(S < S_1)$

ALORS $appar = 0$

SINON $appar = 1$

On calcule ensuite

$$k = \min (p_{car_j})$$

On arrive finalement pour chaque pixel (x,y) à la classification:

(x,y) classe m_1 proba $P''_1 = k_1 * P_1$

classe m_2 proba $P''_2 = k_2 * P_2$

classe m_3 proba $P''_3 = k_3 * P_3$

Le point (x,y) est alors classé dans la classe m_i telle que

$$P''_i \geq \max_k (P''_k)$$

1.4 - APPLICATION

Dans l'application de cartographie de la végétation de la région de "Palni Hills" en Inde, nous avons intégré les connaissances suivantes sur les classes recherchées:

Caractéristiques de type objet (CTO)

crête, vallée, relief modéré, pentes escarpées, plateau, versants nord, versants nord-ouest.....

caractéristiques de type relationnel (CTR)

haut de, fonds de

caractéristiques de type surface (CTS)

reliques

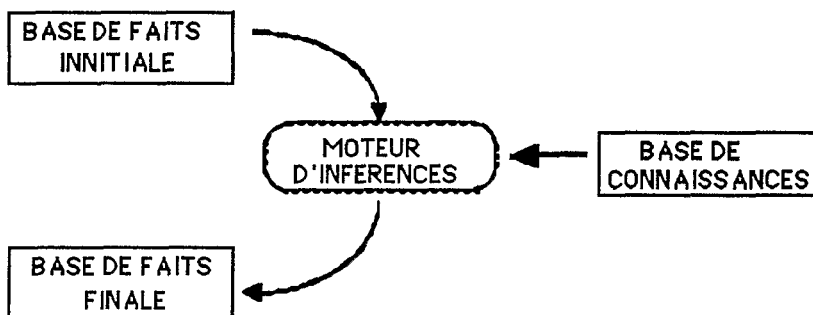
Les caractéristiques générales de chacune des classes étaient rentrées dans un tableau relativement délicat à manier et surtout à modifier dans le cas d'ajout ou de modification de connaissances. Cependant la conception même des caractéristiques générales sous la forme indiquée représentait un premier pas vers une structuration des données proche des systèmes experts. D'où l'idée de passer à la technologie "systèmes experts" pour permettre l'adaptation facile du logiciel à tout type de régions et surtout une manipulation aisée des connaissances.

2 LE SYSTEME ICARE

Le logiciel que nous avons décrit dans le paragraphe précédent a permis de montrer l'intérêt de l'intégration de données exogènes dans la classification pour améliorer notablement les résultats mais il avait des inconvénients majeurs: les données et connaissances utilisées étaient figées dans le code, ce qui fait que la modification de données ou de connaissances devenait un problème informatique délicat. D'autre part les processus déductifs étaient représentés par des algorithmes et formules très empiriques. D'où l'idée d'utiliser la technique des systèmes experts de façon à surpasser ces inconvénients et à transformer le logiciel en un logiciel utilisable dans divers types de régions avec divers types de connaissances et ne nécessitant pas pour l'utilisateur de connaître le logiciel dans son détail. Le type de données et de connaissances mises en jeu restant de même type que celles utilisées dans le logiciel prototype réalisé

On trouve donc dans notre système les différents composants essentiels d'un système expert qui sont

- Une base de faits initiale.
- Une base de connaissances.
- Un moteur d'inférences assurant le raisonnement du système pour aboutir à une base de faits finale



>> La base de faits initiale sera constituée par les éléments suivants:

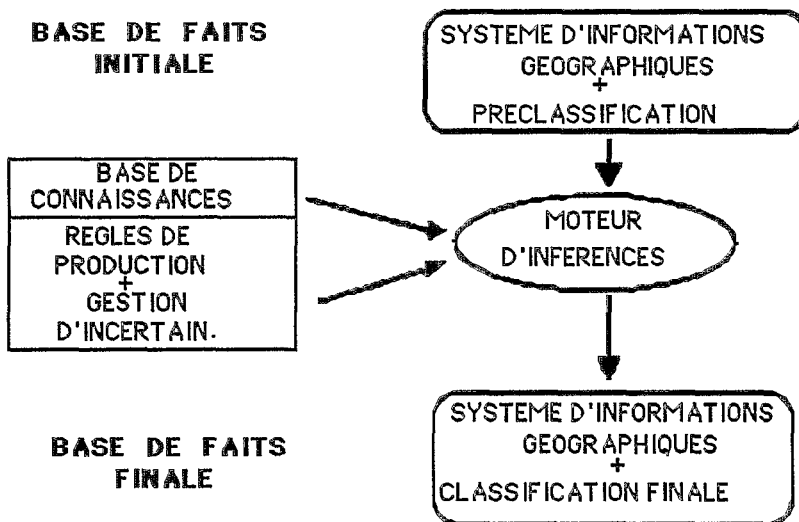
D'une part la préclassification réalisée sur l'image (c'est à dire les facteurs d'appartenance à chacune des classes recherchées, définis pour chaque pixel de l'image) et d'autre part les données exogènes dont on dispose (données brutes du système d'informations géographiques, ou données dérivées de celles-ci par calcul, ou données extraites de l'image par un traitement adapté (ex routes, villages....)).

>> La base de connaissances va en fait contenir sous forme de règles de production (avec gestion de l'incertitude), les connaissances que l'on a sur les classes recherchées dans l'image.

La représentation des connaissances que nous avons mise en oeuvre dans le paragraphe précédent a d'ailleurs été utile pour définir plus facilement les règles de production, et notamment les prémisses de celles-ci.

>> Le **moteur d'inférences** qui va fonctionner en chaînage avant, en appliquant les diverses règles de production de la base de connaissances, jusqu'à permettre d'arriver à la classification finale.

On obtient donc pour le système ICARE le diagramme de base suivant:



2. 1 - DONNEES MANIPULABLES PAR LE SYSTEME

Nous allons décrire maintenant le type de données que le système ICARE va utiliser pour travailler.

On suppose disposer pour chaque pixel que l'on veut classer d'une "préclassification" résultat d'une classification classique. Pour chaque point de coordonnées (x,y) on suppose donc disposer des données suivantes

classe 1	$CF_1(xy)$
classe 2	$CF_2(xy)$
classe 3	$CF_3(xy)$

.....

$CF_i(xy)$ étant le facteur de certitude que le pixel (x,y) appartienne à la classe i (dans notre système cette valeur est supposée comprise entre les deux valeurs -1.0 et +1.0 .

- 1.0: on est sur que le pixel n'appartient pas à la classe i
- +1.0: on est sur que le pixel appartient à la classe i)

De plus le système va utiliser des données qui vont permettre de caractériser le contexte du pixel, contexte qui sera utilisé pour confirmer ou infirmer certains choix de la préclassification. Ces données prendront la plupart du temps la forme "image". Ainsi, si pour la zone donnée on dispose d'un modèle numérique de terrain, on sera capable de connaître l'altitude de chaque pixel (x,y), mais aussi toutes les informations dérivées:

Le point est-il sur un versant ouest ou nord, sur un versant abrupt ou pas, près du fond d'une vallée ou près d'une crête ou soumis à un ensoleillement important ?....Si on connaît l'heure et la date de prise de vues de l'image satellite on pourra dire si le pixel qu'on étudie était à l'ombre (ombre réelle ou portée) ou non

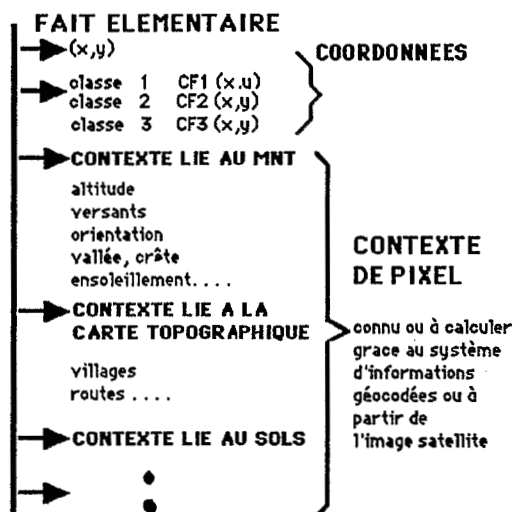
On peut aussi disposer d'une carte des sols donnant pour chaque pixel son contexte géologique, et de cartes topographiques classiques qui, numérisées permettront de disposer notamment de l'information "routes, zones urbaines, canaux...)

Toutes ces données peuvent éventuellement se trouver dans un système d'informations géographiques.

Il se peut que dans certaines zones on ne dispose pas de certaines informations cartographiques parce qu'elles sont absentes ou obsolètes. Alors dans ce cas nous avons prévu de réaliser une analyse de l'image adaptée pour extraire ces informations. Ainsi en est-il du réseau routier et des zones urbaines que l'on peut extraire par des méthodes adaptées à partir de l'image I-HATIR88].

De même la topographie grossière peut être extraite de l'image ILAFFEY82], ou même calculée avec précision sur les images SPOT stéréographiques. Des informations comme le tracé du réseau hydrographique (et éventuellement son débit) vont être obtenues par des méthodes de calcul relativement complexes sur le modèle numérique de terrain [LAMMAL189][ZEHANA90].

Les données manipulables par le système ICARE, qui vont former la base de faits du système expert sont en fait le rassemblement des faits élémentaires décrits ci après:



2.2- INTEGRATION DES CONNAISSANCES DE L'EXPERT

Nous avons déjà indiqué dans le paragraphe 1.3, le type de connaissances que nous pouvons manipuler . Les connaissances doivent concerner les caractéristiques de contexte des différentes classes recherchées. Ces connaissances peuvent présenter un certain degré d'incertitude et d'autre part doivent pouvoir être "vérifiées" sur la base de données dont on dispose (système d'informations géographiques ou informations dérivées, informations extraites de l'image satellite).

Ce sont des connaissances qui s'expriment de fa,con analogue à l'exemple ci-dessous:

Supposons que la classe concernée soit les " sapins ", alors la connaissance des caractéristiques de localisation de cette classe pourraient être:

" Les sapins sont principalement localisés sur les versants Sud à une altitude comprise entre 800 et 1500 mètres "

Dans cette phrase, l'adverbe "*principalement*" montre l'incertitude que l'on a sur la validité de la caractéristique à laquelle on affectera une valeur (ici 0.8).

Quand aux données caractérisant la classe "sapins", elles sont calculables sur le M.N.T.: il s'agit de "versants sud " et "altitude comprise entre 800 et 1500 mètres".

Dans le système ICARE, cette connaissance se traduira par une ou plusieurs règles de production

- La conclusion de la règle étant la classe concernée
- L'incertitude sur la validité de la règle sera l'incertitude de la connaissance.
- Les données calculables sur le M.N.T. dans cet exemple seront les prémisses ou antécédents de la règle.

La règle de production pour représenter les caractéristiques de la classe "sapins" aura donc la forme suivante:

SI	(versants sud) ET (800m < altitude < 1500m)
ALORS	Classe "SAPINS"
CF	=0.8
	règle

CF_{règle} est le facteur de certitude de la règle elle-même et représente la confiance que l'on a dans la règle correspondante (dans nos conventions +1. indique une certitude positive totale et -1. une certitude négative totale).

La base de connaissances construite va donc être un ensemble de règles du type précédent concernant chacune des classes. Cette base de connaissances peut être construite à l'aide d'un compilateur de règles [MINGOT90], logiciel d'interface entre le thématicien qui exprime ses connaissances sur chacune des classes en langage naturel et la base de connaissances elle-même. On propose au thématicien d'exprimer ses connaissances avec des mots-clés prédéfinis qui en fait décriront les différentes caractéristiques définies dans le paragraphe 1.3, mais

on peut aussi lui proposer de définir lui-même ses propres "granules de connaissances" pourvu qu'il fournisse les fonctions permettant de les calculer.

Mots-clés définissant les caractéristiques de type objet, que l'on peut trouver dans la base de données:

Vallée, crête, versant, plateau	...liés au M.N.T.
Route, village, canal,	...liés à la carte topographique
Rocheux, argileux, sol acide,lié à la carte des sols.

Mots-clés définissant les caractéristiques de type relationnel:

Loin de, près de, au dessus de,

Mots clés définissant les caractéristiques de type présence:

Ce sont en fait les mots-clés permettant de décrire l'incertitude qu'on a sur la connaissance, il s'agit d'un codage de l'incertain:

exemple :

seulement (+1.),	principalement (+0.8),	fréquent (+0.4),	rarement (-0.6),	jamais (-1.) ...
---------------------	---------------------------	---------------------	---------------------	---------------------

Mots-clés définissant les caractéristiques de type spatial (dites de niveau 2):

Grandes surfaces, reliques ...

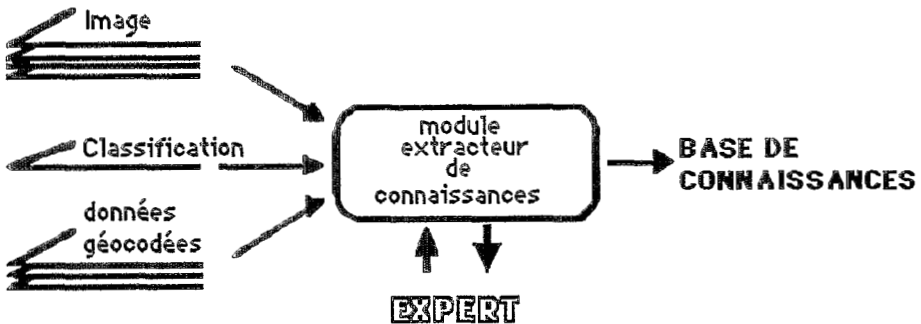
Le compilateur de règles produira donc des règles de niveau 1 (basées sur le contexte du pixel en tant que tel), et des règles de niveau 2 (basées sur des notions de forme . de surface. de relations spatiales avec d'autres objets ou classes).

Voici quelques exemples de formes de règles acceptées par le système:

A et B \rightarrow Cl (CFrègle)	(A ou B) et C \rightarrow Cl (CFrègle)	(A et B) OU C \rightarrow CL (CFrègle)
--------------------------------------	---	---

En fait toute expression dont les opérandes sont des "prémisses élémentaires" et les opérateurs, les opérateurs logiques "et", "ou" sont des prémisses acceptables pour nos règles de production.

Remarquons à ce niveau qu'il est tout à fait raisonnable de concevoir un module capable de construire les connaissances sur les classes par apprentissage. En effet si l'on dispose d'images et des classifications correspondant à ces images, classifications étant considérées comme justes et d'autre part des informations géocodées correspondant à la région. on peut regarder les corrélations existant entre les diverses classes d'intérêt et les prémisses possibles (à partir des caractéristiques spatiales, de type objet et de type relationnelles calculables sur les informations géocodées) . Le degré de corrélation permettra de donner une valeur de certitude à chacune des règles produites.



2.3 - PROCESSUS DEDUCTIF

Le principe de fonctionnement du système ICARE est largement basé sur une manipulation d'informations incertaines. Cette incertitude intervient à plusieurs niveaux

>> A l'état initial, chaque pixel (x,y) de l'image est classé dans chaque classe i avec une certitude $CF_i(x,y)$

Notons ici que l'objectif d'ICARE est justement de mettre à jour ces certitudes $CF_i(x,y)$ en fonction du contexte du pixel et des caractéristiques de chacune des classes.

>> Chaque règle est sure à un certain degré défini par le facteur de confiance dans la règle $CF_{règle}$. On a ainsi choisi les conventions suivantes:

Facteur de confiance

Mots clefs

-1.0	(Faux), jamais, en aucun cas,...
-0.8	(Quasiment faux), très rarement,...
-0.6	(presque faux), rarement,...
-0.4	(assez faux), parfois,...
-0.2	(plus faux que vrai), peu courant,...
0.0	(Ni vrai ni faux), indifféremment,...
0.2	(plus vrai que faux), présent,...
0.4	(assez vrai), commun, fréquent, courant.
0.6	(presque vrai), la plupart du temps,...
0.8	(quasiment vrai), principalement prédominant, surtout, ...
1.0	(Vrai), toujours, seulement, rien que,...

Ces mots clefs et les valeurs de certitude associées doivent être discutées au départ avec l'expert pour accord sur l'échelle des valeurs.

>> Les prémisses (antécédents) des règles, définissant un contexte (favorable ou défavorable) pour une règle de production donnée, vont être calculées pour chaque pixel (x,y) et comme ces prémisses ne sont que rarement totalement réalisées ou totalement non réalisées, il intervient un facteur de certitude de réalisation de la prémisse $CF_{\text{prémisse}}(x,y)$. Notons au passage que toute prémisse est acceptable pourvu que l'on puisse calculer le facteur correspondant.

On est par exemple dans la situation suivante pour un pixel (x_0,y_0) donné:

(x_0,y_0)	classe 1	$CF_1(x_0,y_0)$
	classe 2	$CF_2(x_0,y_0)$

	classe 12	$CF_{12}(x_0,y_0)$

et supposons que la connaissance sur la classe 12 se résume à la règle suivante:

SI (près de village) ou (versants abrupts) ALORS classe 12
($CF_{\text{règle}} = 0.7$)

alors il faut calculer les degrés de réalisation des prémisses

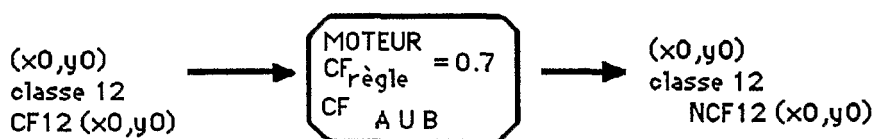
A: "près de village "

et B: "versants abrupts"

on obtient donc deux facteurs de réalisation CF_A et CF_B

on doit alors calculer CF_A ou B à partir de CF_A et CF_B .

Puis le moteur d'inférences réalisera le travail suivant:



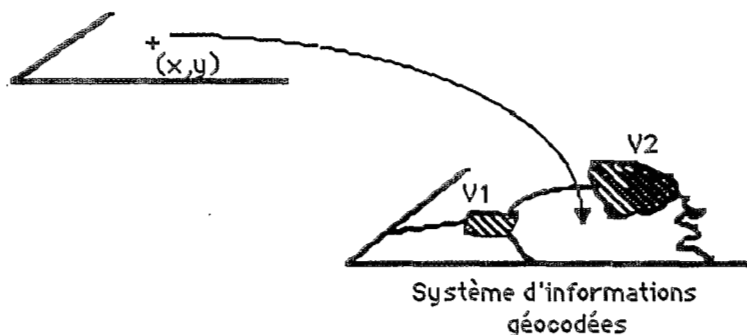
Nous nous bornerons dans la suite à décrire des règles de niveau 1

2.3.1 - Calcul de prémisses

Dans le calcul de prémisses nous avons utilisé suivant les cas, diverses techniques (mesures de distances, théorie des sous-ensembles flous ...).

exemple:

>>> A: "prés de village"



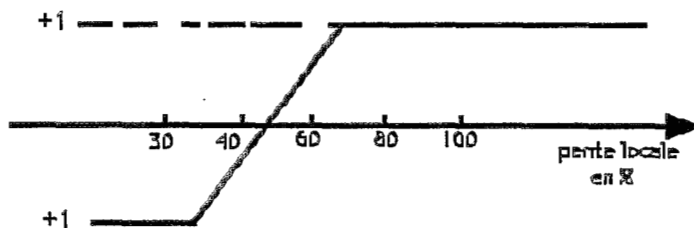
alors

$$CF_A(x,y) = f[d((x,y), V2)]$$

On peut utiliser ici la théorie des sous-ensembles flous, ou mieux, car plus rapide et calculable pour tous les pixels, réaliser une somme d'images dilatées successives (dilatation au sens morphologie mathématique) de l'image binaire des villages seuls [ZEHANA90]. Dans ce calcul de distance nous avons d'ailleurs développé cette technique de dilatations successives en faisant intervenir le relief pour faire des calculs de distances réelles.

>>> B: "versants abrupts"

Pour un point (x,y) on calcule sur le M.N.T., la pente locale en pourcentage, la prémisses "versants abrupts" sera alors plus ou moins vérifiée suivant la valeur de cette pente:



Chaque type de prémisses va en fait impliquer l'existence d'une fonction capable de calculer le facteur de certitude de réalisation de la prémisses correspondante, et ce calcul peut être réalisé à l'avance systématiquement sur tous les points de l'image que l'on veut traiter (utilisation de processeurs parallèles et mémorisation de résultats globaux).

"hauts de crête"	---> fonction_crêtes
"Fonds de vallées"	---> fonction_vallées
"près de routes"	---> fonction_routes

.....

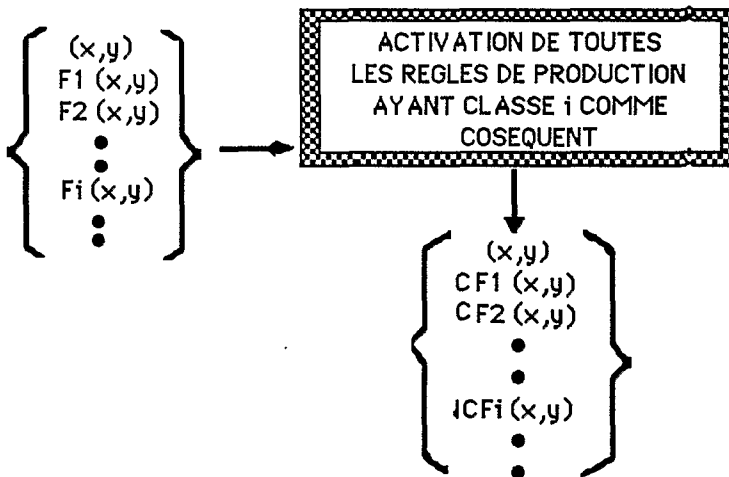
Comment va alors fonctionner le moteur d'inférences ? On part du principe général suivant (qui peut éventuellement être remis en cause pour accélérer les calculs): " Tout pixel peut être classé dans toutes les classes possibles".

Pour accélérer les calculs, on peut définir deux seuils:

Lorsqu'un pixel est classé dans une classe i avec une certitude $CF_i(x,y)$, si $CF_i(x,y) > S_{\text{haut}}$ ou $CF_i(x,y) < S_{\text{bas}}$ alors $CF_i(x,y)$ ne sera plus modifié !!

Le principe de fonctionnement général va donc être le suivant:

Faire évoluer la base de faits et donc les $CF_i(x,y)$



•• CALCUL DE PREMISSES COMBINEES

Suivant que l'on a des "ou" ou des "et", le calcul de la certitude sur la prémisses combinées sera réalisé ainsi

→ Si $A \cup B$ alors...

$$CF_{A \cup B} = \max (CF_A , CF_B)$$

→ Si $A \cap B$ alors...

$$CF_{A \cap B} = \min (CF_A , CF_B)$$

Maintenant que nous avons indiqué comment on procède pour le calcul des prémisses combinées, nous allons détailler le cas de l'activation d'une règle de production où il n'y a qu'une prémisses.

par ex. soit la règle concernant la classe i:

**SI A ALORS Classe i
(CFrègle)**

Règle Ri :
règle n° j
concernant
la classe i

Le facteur de certitude de la règle, indiquant la confiance qu'on a dans la règle (valeur réelle entre -1.0 et +1.0) est en fait ce qui est appelé "coefficient d'atténuation" de la règle pour les règles du système expert MYCIN.

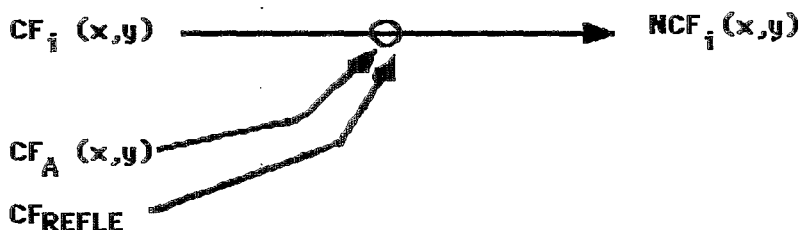
Pour un point (x,y)

>>> Le point appartient à la classe i avec une certitude initiale $CF_i(x,y)$.

>>> La prémisse A est réalisée au point (x,y) avec une certitude $CF_A(x,y)$.

>>> Application de la règle R_{ij} , comportant le facteur de certitude CF_{REGLE} .

On doit donc calculer le nouveau facteur de certitude $NCF_i(x,y)$ que le point (x,y) appartienne ~ la classe i sachant qu'on a appliqué la connaissance inscrite dans la règle R_{ij} .



Ceci est réalisé suivant un algorithme déjà utilisé dans MYCIN [SHORTLIFFE75] pour combiner des résultats de règles différentes concernant une hypothèse donnée (ici cette hypothèse est le classement dans la classe i).

On a déjà le fait "pixel (x,y) classé en classe i avec la certitude $CF_i(x,y)$ " et d'autre part la prémisse A réalisée avec une certitude $CF_A(x,y)$ "atténuée" par le facteur de confiance dans la règle CF_{REGLE} (soit une deuxième certitude "fictive" $CF_A(x,y)*CF_{REGLE}$)

On obtient alors l'algorithme suivant de mise à jour du facteur $CF_i(x,y)$ en appliquant la règle R_{ij} :

SI $CF_i \geq 0$ et $CF_A * CF_{REGLE} \geq 0$

ALORS $NCF_i = CF_i + CF_A * CF_{REGLE} * (1 - CF_i)$

SINON

SI $CF_i < 0$ et $CFA * CF_{REGLE} < 0$

ALORS $NCF_i = CF_i + CFA * CF_{REGLE} * (1 + CF_i)$

SINON

SI $CF_i * CFA * CF_{REGLE} < 0$

et $abs(CF_i * CFA * CF_{REGLE}) < 1$

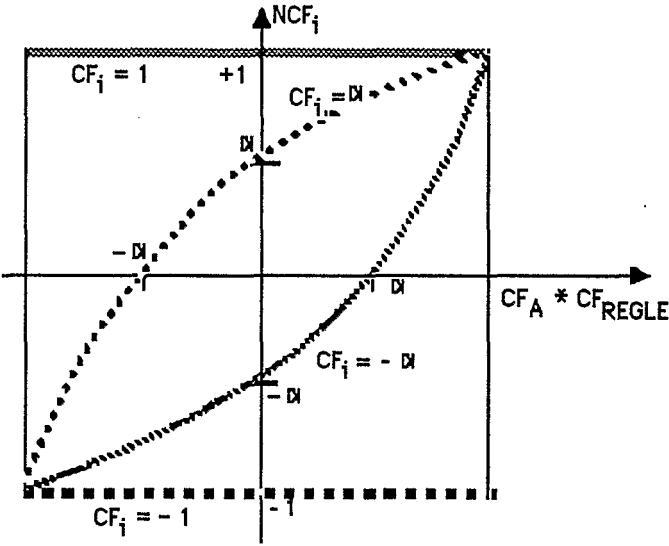
ALORS $NCF_i = (CF_i + CFA * CF_{REGLE}) / (1 - min(abs(CF_i), abs(CFA * CF_{REGLE})))$

SINON $NCF_i = 1$

L'évolution de $CF_i(x,y)$ suivant les valeurs de $CFA(x,y)$ et CF_{REGLE} est donné par le tableau et les courbes ci-dessous:

$CF_A(X,Y)$	CF_{REGLE}	$CF_i(X,Y) \xrightarrow{R_{ij}} NCF_i(X,Y)$
+	+	
+	-	
-	+	
-	-	

Voici les fonctions obtenues pour diverses valeurs de $CF_i(x,y)$:

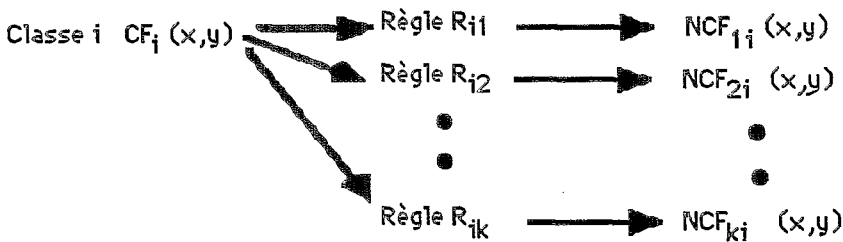


On remarque sur ces courbes que si un $CF_i(x,y)$ est très élevé (voisin de +1.0), il faudra des conditions vraiment défavorables pour faire baisser sérieusement cette valeur, de même si un $CF_i(x,y)$ est très faible (valeur négative par exemple) il faudra des conditions très favorables pour augmenter de façon significative cette valeur.

Ceci permet de justifier le fait que si l'on veut réduire les temps de calcul, on peut décider que si une valeur de $CF_i(x,y)$ est très élevée (supérieure à un seuil donné) ou très basse, on n'applique pas les règles, puisque l'application des règles modifie peu le facteur correspondant.

Pour chaque point et pour chaque classe i , on applique toutes les règles de niveau 1 concernant la classe i (sauf si pour une classe on a décidé de seuiller), on obtient donc autant de nouveaux facteurs de certitude.

Supposons que pour la classe i il y ait k règles applicables:



On appliquera alors l'algorithme commutatif et associatif suivant, décrit pour les règles R_{i1} et R_{i2}

Règle $R_{i1} \rightarrow$ Classe i $NCF_{1i}(x,y)$

Règle $R_{i2} \rightarrow$ Classe i $NCF_{2i}(x,y)$

$$NCF = f(NCF_{1i}, NCF_{2i})$$

SI $NCF_{1i} \geq 0$ et $NCF_{2i} \geq 0$

ALORS $NCF = NCF_{1i} + NCF_{2i} - NCF_{1i} * NCF_{2i}$

SINON

SI $NCF_{1i} < 0$ et $NCF_{2i} < 0$

ALORS $NCF = NCF_{1i} + NCF_{2i} + NCF_{1i} * NCF_{2i}$

SINON

SI $NCF_{1i} * NCF_{2i} < 0$ et $\text{abs}(NCF_{1i} * NCF_{2i}) < 1$

ALORS $NCF = (NCF_{1i} + NCF_{2i}) / (1 - \min(\text{abs}(NCF_{1i}), \text{abs}(NCF_{2i})))$

SINON $NCF = 1$

En appliquant successivement cet algorithme (résultat indépendant de l'ordre dans lequel on traite les valeurs $NCF_{ji}(x,y) \ j=1...k$), on obtient une nouvelle valeur de certitude $NCF(x,y)$

Après application de toutes les règles de niveau 1 pour toutes les classes et tous les pixels, on obtient un ensemble de facteurs de certitude mis à jour:

$$(NCF_i(x,y))$$

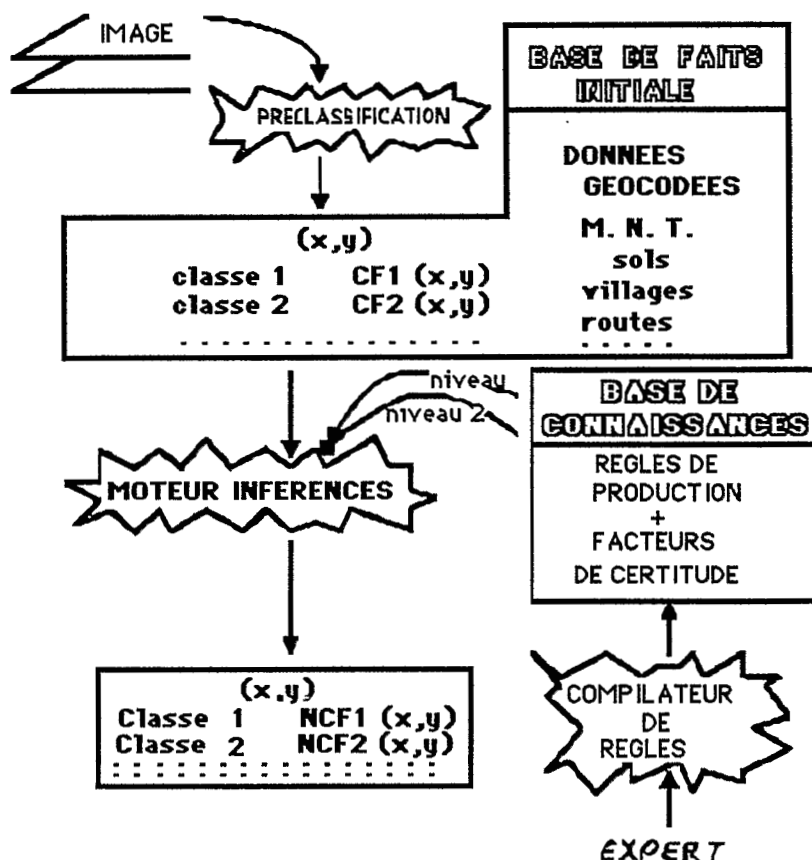
$NCF_i(x,y)$: certitude que le pixel (x,y) soit classé dans la classe i .

Le pixel (x,y) sera alors classé dans la classe k telle que

$$NCF_k(x,y) = \max (NCF_i(x,y))$$

Notons ici que la définition de deux seuils S_{haut} et S_{bas} sur les $CF_i(x,y)$ au delà desquels on ne touchera plus ces valeurs $CF_i(x,y)$ accélérera énormément les calculs et d'autant plus que $(S_{haut}-S_{bas})$ sera faible.

L'ensemble de ce que nous venons de décrire pouvant se résumer dans le schéma suivant:



SCHEMA SIMPLIFIE DU SYSTEME ICARE

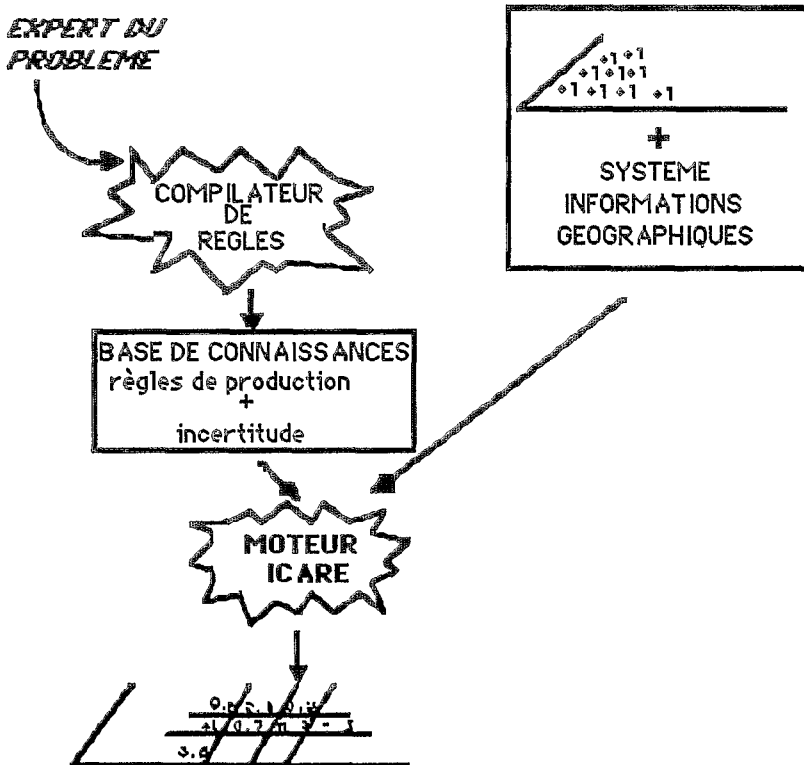
2.4 - I.C.A.R.E. RESOLVEUR DE PROBLEMES

Le système ICARE peut aussi être utilisé en résolution de problèmes, à condition que les données manipulées pour résoudre le problème soient acceptables pour ICARE (en fait ces données doivent être calculables sur le système d'informations géographiques ou extraites de l'image).

D'autre part la solution du problème doit apparaître comme une carte des points où le problème posé sera plus ou moins résolu. Les points de valeur +1.0 étant ceux qui "résolvent totalement" le problème, ceux de valeur -1 ceux qui ne peuvent en aucun cas le résoudre.

La base de connaissances va en fait être l'ensemble des connaissances définissant le meilleur contexte calculable pour résoudre le problème. Bien entendu au départ du système tous les points sont censés résoudre le problème avec la même efficacité, chaque point est donc affecté initialement de la valeur +1.

Le fonctionnement en résolveur de problèmes peut être représenté par le schéma suivant:



On a donc là une "retombée" intéressante pour les systèmes d'informations géographiques du système ICARE .

2. 5 CONCLUSION

Le système proposé permet d'améliorer nettement les résultats de classification (dans l'exemple de la zone de Palni hills, Inde).

Il intègre d'une part des connaissances normalisées (courantes) liées au MNT et aux cartes topographiques courantes. mais d'autre part il est très ouvert à l'intégration de connaissances nouvelles.

La modification et l'adjonction de connaissances devient aisée pour l'utilisateur.

Dans la mesure où la résolution spatiale des images satellite va s'améliorer, notre équipe va mettre en oeuvre des techniques de recherche d'objets et d'analyse globale des "paysages" analogues à celles utilisées dans l'interprétation d'images aériennes [MCKEOWN89].

BIBLIOGRAPHIE

- [CAROPP86] Caroppo C., Di Gennaro V., Pasquarallo G. "UMUS: A demonstration project for usage of LANDSAT MSS and ancillary data in land cover mapping of large areas in southern Italy "proceedings of IGARSS'86 symposium ZURICH (1986)
- [DESACHY80] Desachy J. "Texture features in remote sensing imagery" in DIGITAL IMAGE PROCESSING. NATO-ASI series . BONAS (1980)
- [DESACHY86] Desachy J. . Castan S., Fisse G. "Introduction of thematician knowledge in remote sensing imagery interpretation" symposium Commission IV of ISPRS "Mapping from modern imagery" Edinburgh (1986)
- [DESACHY88] Desachy J., Debord P., Castan S. "An expert system for satellite image interpretation and GIS based problem solving" 16 th international congress of ISPRS Kyoto (1988)
- [DESACHY89(1)] Desachy J., Lammali N. "Automatic hydrological network computation from digital elevation model" IGARSS'89, IEEE 89 CH 2768-0 ,Vancouver (1989)
- [DESACHY89(2)] Desachy J., Debord P. " I.C.A.R.E.: An expert system for automatic mapping from satellite imagery" IEEE international conference on image processing ICIP'89 SINGAPORE september (1989)
- [DESACHY89(3)] Desachy J. "ICARE: Système expert d'interprétation d'images satellite et de résolution de problèmes liés aux systèmes d'information géographique" table ronde internationale CNRS Télédétection et paysages tropicaux CNRS-CGET Bordeaux octobre (1989)