

COLLOQUE NATIONAL SUR LE TRAITEMENT DU SIGNAL ET SES APPLICATIONS

NICE du 26 au 30 AVRIL 1977

RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

J. C. SIMON

Institut de Programmation, Université Paris VI, 4, Place Jussieu 75005 PARIS

RESUME

La Reconnaissance des Formes (R.F.) est née avec l'arrivée des ordinateurs de troisième génération, qui rendaient possible le traitement et la reconnaissance des signaux et des images par des techniques algorithmiques.

Le but de la R.F. est d'interpréter à l'aide d'ordinateurs les structures de données fournies par des capteurs physiques. Le plus souvent cette interprétation se borne à une "identification". Il s'agit donc plus précisément de trouver une forme constructive d'une fonction d'identification \mathcal{E} , appelée opérateur d'identification, permettant de passer d'une structure de données d'entrée à une représentation interprétable par un nom.

Les informations disponibles pour construire \mathcal{E} sont un ensemble fini dit d'apprentissage et des propriétés de l'espace des représentations d'entrée ou des propriétés de l'espace fonctionnel des opérateurs d'identification.

Propriétés de l'espace des représentations. Ces propriétés topologiques sont exprimées par des fonctions de dissemblance ou des distances. Elles permettent d'obtenir des "descriptions symboliques": analyse en composantes principales, hiérarchies, arbre de longueur minimum. Une technique particulièrement utilisée est celle des partitions adaptatives (clustering). L'invariance d'une représentation à certaines transformations permet de réduire la dimension de l'espace des représentations.

Propriétés des opérateurs d'identification. Les propriétés de certains algorithmes permettent de guider leur emploi en R.F.: opérateurs linéaires adaptatifs ou non - opérateurs logiques - opérateurs linguistiques. D'une façon générale la puissance des fonctions récursives peut être utilisée à condition de maîtriser leur sémantique. Des fonctions d'évaluation ou heuristiques, l'information mutuelle de Shannon en particulier, permettent de guider les choix.

Enfin les méthodes développées en Intelligence Artificielle s'appliquent au domaine de la reconnaissance et de l'interprétation de dialogues de question-réponse avec un système informatique. Le but est encore à partir d'un ensemble de questions de construire automatiquement le programme qui fournira les réponses correctes.

SUMMARY

Pattern Recognition (P.R.) is a domain born with the third generation computers, which made possible the algorithmic treatment and recognition of signals and images.

The goal of P.R. is to interpret with computers the structured data given by sensors (input features). Most of the time it is an "identification". More precisely, a constructive form, the identification operator, must be given to a function \mathcal{E} of domain the datas, of values the names.

The information available to construct \mathcal{E} are a finite learning set of examples, and the properties of the input feature space and/or the properties of identification operators.

Properties of the input feature space: These topological properties are given by dissemblance functions or distances. They allow to obtain symbolic descriptions: factorial analysis, hierarchies, minimum spanning tree. Clustering is also a technique leading to optimum classifications. The invariance of a representation to transformations may lower the dimension of the feature space.

Properties of identification operators: the properties of an algorithm may guide his use in P.R.: linear operators adaptive or not, logical operators, syntactic operators. More generally recursive functions should be used, if their semantic is known. Heuristic may guide the automatic build up of P.R. operators. The Shannon mutual information is used.

The techniques of Artificial Intelligence apply to the domain of understanding natural language and the construction of intelligent question answering systems. From a training set of question and answer, again the goal is to build up a program which from the question will answer the proper answer.



RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

§ 1 RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

1-1 Le Traitement du Signal est un domaine scientifique né avec l'Electronique. Il s'est particulièrement développé en vue d'applications comme la téléphonie, la télévision, le radar, le sonar, disons de façon générale les vastes domaines des Communications et de la Mesure. Il a fait appel à des techniques mathématiques particulièrement raffinées, la statistique, le calcul différentiel et intégral, la topologie.

Cependant l'analyse ultime du signal était laissée presque toujours à l'être humain.

L'arrivée des ordinateurs de la troisième génération a changé le panorama pour certains scientifiques et ingénieurs, qui ont voulu voir dans ces nouvelles machines un moyen différent de traiter les signaux et, pourquoi pas, un moyen de se substituer à l'homme dans ses fonctions de reconnaissance les plus raffinées.

La technologie mettait en évidence plusieurs moyens nouveaux:

Des *mémoires rapides* de grande capacité, pratiquement ignorées des électroniciens, il y a vingt ans seulement.

Des *opérateurs arithmétiques* et logiques très rapides, permettant de faire un nombre immense d'opérations par secondes.

Et peut-être le plus important, la possibilité de *simuler* n'importe quelles machines dans une classe extrêmement large: les machines de Turing, ou plus restreinte les automates simples ou à pile, dont l'équivalent mathématique est la classe des fonctions récursives.

Les notions plus pratiques de "systèmes informatiques" se sont dégagées, permettant de mieux mettre en oeuvre les idées un peu trop générales de la "Turinologie".

Qu'on permette à un ancien de l'électronique d'évoquer le temps où son laboratoire était encombré de cadavres de châssis, reliques d'essais infructueux, mais hélas souvent combien laborieux. Il est évidemment plus facile d'écrire un programme et de le modifier, et c'est surtout beaucoup plus rapide et moins coûteux.

Ce n'est qu'assez récemment que l'intérêt de la machine générale sur la machine spécialisée est apparu clairement, du moins en recherche. Cependant qu'on ne se méprenne pas sur le sens de cette déclaration. L'auteur est bien conscient de l'intérêt pour certains problèmes de calculateurs ou transformateurs spécialisés, optiques par exemple. D'ailleurs l'électronique des circuits retrouve toutes ses prérogatives lorsqu'il s'agit d'obtenir une machine optimale.

C'est donc pour ces raisons qu'un groupe de scientifiques et d'ingénieurs ont voulu aborder les problèmes de traitement de signal et particulièrement de reconnaissance avec les ordinateurs. Ils forment maintenant une communauté scientifique internationale la "Reconnaissance des Formes" (Pattern Recognition) qui depuis 1969 tient des Symposium bisannuels: les International Joint Conference on Pattern Recognition, IJ CPR, qui réunissent plus de 500 participants. Une société internationale vient d'être créée en 1976 l'I.A.P.R., affiliée à l'I.F.I.P. dont elle sera la section de Reconnaissance des Formes. Cette communauté est en liaison étroite avec celle de l'"Intelligence Artificielle" dont les Symposium I.J.C.A.I. connaissent un développement équivalent.

Pourquoi et comment ces communautés se sont différenciées des électroniciens du Traitement de Signal et des théoriciens de la Théorie de l'Information.

Tout d'abord par la conviction que *les ordinateurs n'étaient pas seulement un moyen nouveau*, mais que ce nouveau moyen exigeait un nouveau langage et par là même une *nouvelle philosophie*.

Donc il ne s'agissait pas de *transposer* (plus ou moins facilement) *les théories de l'Information* exprimées dans le langage mathématique habituel, mais de s'exprimer dès l'abord dans le langage des algorithmes, d'y *formuler directement les problèmes posés*, de tenir compte des nouvelles théories informatiques des automates, des langages formels, des fonctions récursives, des graphes, de la complexité.

En particulier un signal temporel ou une image ne se représente plus par une fonction continue à une ou deux variables, mais par une suite d'échantillons numériques. Les transformations tiennent compte du caractère discret et fini des représentations. Un calcul est considéré comme un algorithme et on tente de montrer ses conditions de validité, parfois très différentes des conditions de validité mathématiques.

Enfin un algorithme est évalué d'après les performances en machine. Certains calculs ont un caractère "exponentiel", d'autres "polynomiaux". La menace de l'"explosion combinatoire" pèse sur l'informaticien. Il cherche donc à découvrir *directement* des algorithmes simples et performants, plutôt que de transposer des techniques peut-être valables "en principe", mais inapplicables en réalité parce que trop coûteuses en mémoire et en temps de calcul.

Nous reportons le lecteur à la communication "Utilisation de techniques de reconnaissance de formes sur le signal de parole" dans cette même réunion du GRETSI. Nous croyons pouvoir affirmer que la philosophie décrite plus haut a permis d'obtenir *en temps réel sur mini-ordinateur*, en synthèse et reconnaissance de la parole, des performances analogues sinon meilleures à celles obtenues plus classiquement sur du matériel spécialisé tel des vocoders et en utilisant de très gros ordinateurs de la classe PDP 10, IBM 370 ou CDC 6.600. De la même manière la communication "Extraction de primitives dans des images de point", montre l'application de la philosophie de la reconnaissance des Formes à des traitements d'images. Le lecteur familier des techniques de traitement de signal ne manquera pas d'en saisir les aspects originaux.

Nous nous proposons maintenant d'exposer les grandes lignes de la "Reconnaissance des Formes". Ses buts sont de *rechercher les algorithmes qui, traduits sous la forme de programme, permettent, à partir des données de capteurs, de reproduire efficacement les phénomènes de perception et de compréhension de l'être humain*.

§ 2 DEFINITIONS

2-1 Un *capteur* de mesure joue un rôle analogue à un organe des sens. Il délivre un certain nombre de résultats de mesures.

Ces résultats sont habituellement nommés *données initiales* ou *description initiale*. Ils comportent une *représentation* et son *interprétation*. Par exemple la description initiale d'un objet peut être représentée par une *liste* (ou vecteur) de valeurs numériques ou non.

$$X = (x_1, x_2, \dots, x_n)$$

Une telle donnée peut être dite "structurée" dans la mesure où le numéro de la liste correspond à un résultat interprétable.

Une représentation peut être un ensemble de signes interprétables, par exemple la suite des symboles $X = (x_1 \dots x_n)$. En informatique une représentation est

le plus souvent un ensemble d'états de cellules de mémoire.



RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

Il est clair qu'un ensemble d'états en mémoire n'a aucune "signification" si l'on n'est pas capable de l'interpréter: par exemple tel état de cellule de mémoire à telle adresse représente telle intensité de lumière sur telle photodiode d'une mosaïque à tel instant. Nous appelons *interprétation triviale* une telle interprétation.

Les informaticiens donnent souvent le nom de *structures de données* à une représentation en mémoire dont les éléments sont reliés entre eux par des pointeurs. Il en résulte des structures de graphe, dont la liste est un exemple particulier. Les liaisons entre éléments peuvent bien entendu être interprétées.

Par définition nous appelons *entité d'information* le couple formé par une représentation et une de ses interprétations. Cette notion ne doit pas être confondue avec la *quantité d'information* qui n'est autre que le volume de mémoire en bits, nécessaire à une représentation informatique.

En fait l'interprétation d'une représentation est une notion difficile à limiter donc peu précise. Les mathématiciens eux-mêmes admettent que le nombre des propriétés d'un "objet mathématique" peut ne pas être borné. Ainsi le nombre représenté par 1327 a des propriétés en nombre infini dès qu'on le compare à tous les autres nombres. Seules un certain nombre de propriétés sont "intéressantes" pour un certain but, par exemple la divisibilité pour prouver que le nombre est premier ou non. Cet ensemble de propriétés est à coup sûr fini, sinon comment démontrer des théorèmes. L'interprétation d'une représentation X peut de même être infiniment variée. Parmi celles-ci seules quelques-unes sont "utiles". A coup sûr la première, que nous qualifions de triviale, est de savoir à quelle mesure, à quel phénomène on a affaire. L'interprétation finale pouvant être précisément une *identification*, le nom d'un objet dans un ensemble d'objets possibles.

Mais d'autres interprétations peuvent être désirables: ainsi la description d'une scène, une action, la modification d'une structure de données et bien d'autres...

Ces interprétations correspondent en Reconnaissance des Formes à un traitement plus ou moins automatique de la représentation en mémoire, traitement qui peut être fort complexe.

De la même façon, on ne saurait séparer la chaîne de caractères d'un langage de programmation, représentation d'un type particulier, avec l'interprétation qui en est faite par le programmeur ou celle de la machine.

L'interprétation du programmeur est un certain algorithme. L'interprétation de la machine est l'exécution d'un algorithme qui, comme l'enseigne l'expérience, est bien souvent différent de celui qui est imaginé par le programmeur.

Dès 1910, le linguiste F. DE SAUSSURE [50] avait insisté sur l'inséparabilité du *signifiant* (la représentation) et du *signifié* (son interprétation). Le couple signifié-signifiant forme un *signe* (l'entité d'information).

La *sémantique* d'un signe, d'un discours est l'ensemble des interprétations, des propriétés du signe ou du discours.

Il serait bien entendu tout à fait irréaliste de chercher à produire par un traitement automatique toutes les interprétations possibles d'un ensemble de représentations. La plupart des efforts de Reconnaissance des Formes se borne à *identifier*, c'est-à-dire à séparer en classes disjointes l'ensemble des représentations possibles.

Une interprétation possible d'une représentation ou structure de données est une *forme caractéristique* ou simplement *forme*, en anglais *feature*. Ce terme correspond le plus souvent à une interprétation humaine. Ainsi si nous examinons des représentations de caractères imprimés nous distinguerons des formes (features) telles que des segments de droite, des points de rebroussement, des courbes concaves ou convexes, etc... Par extension le résultat de toute interprétation partielle ou totale peut être une forme. Il arrive même que les données des capteurs soient appelées formes (features) élémentaires. Le résultat d'un programme de reconnaissance sera aussi une forme (feature). S'il est final nous l'appelons un nom, s'il est partiel nous disons que nous avons reconnu une *forme caractéristique* de l'objet à reconnaître.

§ 3 FONCTIONS ET OPERATEURS DE RECONNAISSANCE

3-1 Devant la complexité des problèmes de reconnaissance d'image, de parole, de données médicales, physiques, économiques, les efforts concrets de Reconnaissance des Formes se sont bornés

- soit à obtenir une meilleure représentation des données expérimentales, afin de permettre à l'observateur humain de les interpréter.
- soit à présenter des opérateurs d'identification dont le résultat est de séparer les représentations en classes disjointes.

Afin de fixer les idées, prenons l'exemple d'un chercheur en médecine examinant le tableau d'un ensemble de malades présentant des anomalies sanguines.

Soit X_i la liste de mesures correspondant au patient i .

$$X_i = (x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{in})$$

Le tableau \mathcal{T} des mesures est constitué par l'ensemble des n mesures sur les m patients. La j ème mesure est par exemple le taux d' α -globuline. Si m et n sont grands le tableau est "inutilisable" tel quel.

Les techniques de Reconnaissance des Formes vont permettre:

- soit d'obtenir une nouvelle représentation: par exemple représenter chaque malade par un point dans un plan (analyse fonctionnelle, hiérarchie et autres)
- soit d'obtenir un programme de classification de ses malades en maladies connues: leucémie, infection, etc...

Nous nous bornons dans cet exposé à de telles techniques effectives.

Le terme d'*identification* est réservé au dernier cas:

Soit \mathcal{X} l'ensemble des représentations possibles fournies par les capteurs, X une représentation. Ces représentations sont appelées données (ou structures de données) initiales, ou encore espace des formes initiales (initial feature space).

Soit Ω l'ensemble des noms identifiant les objets, ω un nom d'objet. Le but d'une reconnaissance est de trouver l'application ou fonction \mathcal{E} , sur l'espace \mathcal{X} et à valeur dans Ω .

$$\mathcal{E} : \mathcal{X} \rightarrow \Omega$$

Le problème essentiel de la Reconnaissance des Formes est de trouver une *forme constructive* de cette application. Inversement un programme réalisant effectivement une telle application est un programme d'identification. Mais cela n'implique nullement que cette identification ait une signification. Ce n'est que dans la mesure où l'interprétation fournie par le programme *coïncide avec l'interprétation humaine*, que nous pouvons affirmer que la reconnaissance est correcte.



RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

Donnons un *exemple*: soit un réseau de 10×20 d'éléments pouvant être dans l'état blanc "0" ou noir "1". Un tel réseau permet de représenter les 27 lettres de l'alphabet (y compris l'absence de lettre). Une représentation X est la suite des 200 valeurs 0 ou 1, correspondant aux états des éléments du réseau, balayé de gauche à droite et de haut en bas. Interprétons une telle représentation comme un nombre positif. Divisons-le par 27. Le résultat obtenu est un chiffre compris entre 0 et 26. Une application constructive est ainsi obtenue entre l'ensemble des représentations de mesure et l'ensemble des lettres. Il est clair qu'un tel algorithme n'a aucune chance d'effectuer une reconnaissance coïncidant avec notre observation.

DEFINITIONS

Différents noms équivalents sont donnés aux formes *constructives* d'une fonction de Reconnaissance des Formes

- *Algorithme* ou *procédure*, si elle est interprétée dans le domaine des mathématiques informatique par exemple comme un schéma de programme

- *Programme*, si elle est interprétée dans le domaine des machines logiques, par exemple microprogrammées.

Nous avons déjà défini une fonction d'identification comme une application \mathcal{E} de l'espace X des représentations X dans l'ensemble Ω des noms. Cette notion est abstraite et ne correspond pas à une procédure constructive. Cette application peut être effectivement réalisée par un programme ou ce qui revient au même par une machine logique par exemple microprogrammée.

Une fonction "mathématique" ou "application" peut ne pas être définie constructivement, c'est-à-dire interprétable par une procédure. Il est courant en mathématique qu'une fonction soit *caractérisée par ses seules propriétés*, par exemple la continuité, la dérivabilité etc... Les mathématiciens raisonnent sur un tel objet abstrait, trouvent et démontrent de nouvelles propriétés, sans jamais le construire effectivement.

La démarche informatique est inverse. Il est fréquent que l'on construise une procédure et que l'on cherche après coup ses propriétés. C'est là l'objet d'une activité des théoriciens de l'informatique, la sémantique des programmes, cf. MANNA [51].

La construction d'une procédure, d'un opérateur d'identification correct relève de la même approche: *trouver une procédure réalisant une application ayant telle ou telle propriété*. La solution d'un tel problème n'est nullement assurée. Beaucoup d'efforts sont faits en informatique théorique sinon pour le maîtriser, du moins pour mieux l'appréhender.

Un spécialiste de reconnaissance doit en être conscient. En particulier, il est absolument essentiel qu'il soit capable au moins de *dégager et définir clairement à quelles propriétés doit satisfaire la fonction d'identification*. La seconde étape consistera à *trouver une procédure réalisant effectivement cette fonction d'identification*.

Il est fréquent que l'approche informatique courante soit faite: le "reconnaisseur" est convaincu que tel algorithme de reconnaissance sera satisfaisant, il cherche à l'adapter au problème posé. Cette approche intuitive donne bien souvent de grandes déceptions...

L'expérience montre qu'en gros les fonctions d'identification peuvent prendre l'une des trois formes suivantes caractéristiques:

- 1) une identification en mémoire
- 2) l'utilisation de fonctions caractéristiques
- 3) l'utilisation de fonctions récursives

3-2 IDENTIFICATION EN MEMOIRE

Une mémoire adressable composée de 2^a cellules de b bits, permet de définir effectivement une application d'un ensemble A de 2^a éléments dans un ensemble B de 2^b éléments. Il suffit qu'à chaque adresse définie par a bits, corresponde une cellule de b bits.

Ce type de fonction dite "en table", peut définir constructivement une fonction d'identification. Il suffit que l'état de la cellule soit interprétable comme le nom correct ω . L'ensemble T d'apprentissage peut être ainsi implémenté. Malheureusement une telle fonction n'est pas définie en dehors du domaine A , en général une très faible partie du domaine X intéressant.

3-3 FONCTIONS CARACTERISTIQUES

Soit X un objet. A chaque forme de nom ω_i , correspond une fonction g_i . L'algorithme d'identification est alors:

$$g_i(X) > g_j(X), \quad \forall j \neq i \text{ entraîne l'affectation}$$

de X à ω_i ; ou la même relation avec le signe $<$.

Dans le premier cas une fonction g doit être *interprétée* comme une *fonction de ressemblance à la forme de nom ω_i* , dans le second une *fonction de dissemblance*.

DENSITE DE PROBABILITE

Tout un groupe de recherche de Reconnaissance des Formes s'intitule Reconnaissance des Formes *statistique*. Ces recherches utilisent une terminologie probabiliste. En particulier, à partir de certaines données, il s'agit de construire une fonction dite de *densité de probabilité*.

En réalité une telle fonction $P(X(\omega_i))$ joue le rôle d'une fonction caractéristique et n'est autre qu'une *fonction de ressemblance*. Le discours probabiliste n'ajoute rien à l'algorithme lui-même, qui est exactement celui décrit plus haut.

Deux groupes de techniques permettent d'établir de tels $P(X(\omega_i))$.

i) les techniques *paramétriques*: la fonction est choisie a priori, par exemple une gaussienne, et ses paramètres sont ajustés au problème.

ii) les techniques *non paramétriques*: la fonction est établie par des procédés qui rappellent l'interpolation. Citons les histogrammes, l'emploi de fonctions orthogonales, les estimateurs de PARZEN, ROSEMBLATT, et de LOFTSGARDEN et QUESENBERRY cf COVER [3] ou KANAL [52], enfin les techniques fondées sur les N plus proches voisins.

FONCTIONS POTENTIELLES

Un noyau A_i de points représentatifs est choisi, qui peut d'ailleurs être réduit à un seul point, alors

$$g_i(X) = \sum_{X' \in A_i} K(X - X') \mu(X') \quad (1)$$

$K(X - X')$ est une fonction de ressemblance, μ un coefficient. La formule (1) est à rapprocher de la formule d'un potentiel

$$g_i(X) = \int_{A_i} K(X, X') d\mu(X') \quad (2)$$

$K(X, X')$ fonction symétrique décroissante avec la norme $\|X - X'\|$, μ étant une mesure sur l'espace X .

Si $K(X, X')$ est de la forme $K(X - X')$, la formule (2) devient une formule de convolution, justifiable d'un traitement par Transformation de Fourier.

Les g_i sont des fonctions de ressemblance.



RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

FONCTIONS D'INERTIE

Mêmes formulations que (1) ou (2) mais $K(X, X')$ est une fonction de dissemblance, en général une distance. Les g_i sont des fonctions de dissemblance.

Une dissemblance à une représentation particulière peut ainsi être estimée, nous en verrons des exemples nombreux dans le paragraphe 4-2.

FONCTION AD HOC

Le type des fonctions g_i est choisi a priori, sans autre justification que sa simplicité, voir par exemple DUDA et HART [9] chapitre V.

Différentes techniques itératives permettent d'ajuster les paramètres afin d'obtenir des résultats corrects sur un ensemble d'apprentissage.

Un fonctionnement correct sur X suppose implicitement que l'espace X est séparable, l'expérience prouve le plus souvent qu'il n'en est rien.

Pour conclure ce sujet des fonctions caractéristiques, soulignons que le discours qui accompagne les choix est le plus souvent très artificiel.

Il importe de bien se persuader qu'un tel ensemble de fonctions permet simplement d'évaluer une mesure de ressemblance ou de dissemblance par rapport à un certain nombre de formes de noms ω . L'algorithme est bien le même, quel que soit le discours qui le présente... Il n'existe pas ici de différence "opérationnelle" entre les concepts de densité de probabilité, fonctions potentielles, distances généralisées.

3-4 FONCTIONS RECURSIVES

Les procédés exposés en 3-2 et 3-3 utilisent des formes très primitives de fonctions constructives. En réalité la classe disponible de fonctions constructives la plus large est la classe des fonctions récursives. C'est-à-dire les fonctions calculables, qu'on peut toujours représenter par des "programmes".

Ainsi les opérations algébriques ne sont plus les seules utilisées comme en 3-3. Il est donc possible de ranger dans la classe présente l'utilisation d'algorithmes ou opérateurs algébriques, logiques, linguistiques, conditionnels et leurs combinaisons récursives ou non.

Les opérations logiques et conditionnelles ont été naturellement introduites par les techniques de questionnaire cf PICARD [28]

Les opérations linguistiques, essentiellement l'emploi de règles de réécriture, sont couramment utilisées pour la reconnaissance de certains signaux médicaux STOCKMAN [53] de parole HATTON [18] d'images FU [54] GIPS [16].

L'opérateur d'identification d'une forme caractéristique (feature) ou totale (un nom) peut être construit par combinaison d'opérateurs élémentaires.

Signalons l'emploi d'heuristiques afin d'évaluer les performances des opérateurs ainsi construits (feature evaluation) cf SABAH [33], ROCHE [31], KANAL [52] p. 703.

Signalons enfin l'intérêt de la récursivité. Ce concept, déjà utilisé dans les grammaires de formes, permet de représenter sous forme très compacte des opérateurs qui peuvent engendrer ou reconnaître des structures potentiellement infinies. L'emploi de programmes récursifs commence à être systématiquement utilisé dans les recherches de reconnaissance linguistique cf JOUANNAUD [21].

Nous allons examiner maintenant comment un algorithme de Reconnaissance des Formes peut être établi à partir des propriétés du problème posé:

Posons-nous donc la question fondamentale: *quelles sont les informations utilisables permettant de construire un algorithme de Reconnaissance des Formes?*

Cet ensemble d'informations ou sémantique du problème consiste en:

un ensemble fini dit d'apprentissage et des propriétés de l'espace X des représentations ou des propriétés de l'espace fonctionnel des opérateurs de Reconnaissance de Formes.

REMARQUE

En principe une forme (feature) ne devrait pas être séparée de son opérateur de reconnaissance. Les formes élémentaires d'entrée, que nous avons appelées données initiales ou descriptions initiales, sont les résultats donnés par des opérateurs physiques, les capteurs. En principe, sauf pour la commodité, il n'est pas justifié de leur attribuer une qualité particulière. Ces données seront utilisées par d'autres opérateurs, qui à leur tour, donnent des résultats, appelés formes par le reconnaisseur.

Il est important de séparer les formes idéales dont nous parlons couramment; un segment de droite, une courbe par exemple, du résultat obtenu par un programme et que nous cherchons à interpréter comme la représentation d'une forme idéale.

Parfois il n'est pas possible d'interpréter le résultat partiel obtenu par un opérateur de reconnaissance:

Exemple: les meilleurs programmes actuels de reconnaissance de mots parlés, KOVALEVSKI, ZAGORUJKO en URSS utilisent un certain nombre de formes types, de sonogrammes, de l'ordre d'une centaine, pour reconnaître des sons élémentaires. Si certains peuvent être identifiés à des voyelles, la plupart ne peuvent être interprétés comme des "phonèmes", sons atomiques des phonéticiens.

§ 4 INFORMATIONS ET METHODES POUR CONSTRUIRE UNE FONCTION DE RECONNAISSANCE FONDEES SUR UNE TOPOLOGIE DE L'ESPACE DES REPRESENTATIONS

4-1 ENSEMBLE D'APPRENTISSAGE

Il est habituel que parmi les données d'un problème de Reconnaissance des Formes, plus précisément d'identification, un ensemble T d'apprentissage soit connu a priori. L'ensemble T est un ensemble fini de couples (X, ω) , $X \in E \subset X$ et $\omega \in \Omega$.

Cet ensemble T définit un "échantillonnage" de la fonction ξ d'identification.

Etant donnée une procédure P acceptant les X pour données d'entrée et fournissant ω pour données de sortie, la première exigence est bien sûr que T fasse partie du graphe de l'ensemble des (X, ω) fourni par P .

Le terme d'apprentissage provient de ceci: soit $S(\lambda)$ une procédure dépendant d'un paramètre multidimensionnel λ , choisie comme procédure d'identification. On peut penser à choisir λ de façon que la dite procédure ait T dans son graphe. Soit $S(\lambda_0)$. Partant d'un λ quelconque certains procédés itératifs peuvent permettre d'aboutir à un λ_0 . Ces procédés ne sont pas sans rappeler un apprentissage. L'algorithme dit du Perceptron en est un exemple, NILSSON [26]. Malheureusement une fonction, définie sur tout un domaine de X , ne peut être déterminée uniquement par un nombre fini de couples, à moins de spécifier très particulièrement sa nature. Ce problème est familier aux ingénieurs qui cherchent à interpoler un signal échantillonné monodimensionnel.

La solution adoptée pour le problème multidimensionnel du Perceptron était simplement de considérer une fonction linéaire à n paramètres et de choisir ces n paramètres de façon à satisfaire les conditions sur l'ensemble T d'échantillonnage. L'expérience a montré qu'une telle façon d'opérer amenait à de graves déconvenues. Aucune garantie de bon fonctionnement n'existe en dehors de l'ensemble T . Le problème fondamental de la Reconnaissance des Formes est précisément



RECONNAISSANCE DES FORMES ET TRAITEMENT DU SIGNAL

d'obtenir une "extension" de \mathcal{E} en dehors de T . C'est le difficile problème de la généralisation. SIMON [38].

REMARQUE

Certains problèmes peuvent *comporter* ou *ne pas comporter* la donnée d'un ensemble T d'apprentissage. On dit parfois qu'on a à construire une fonction de reconnaissance respectivement *avec* ou *sans professeur*.

Il est clair que la donnée de quelques exemples ne peut suffire à déterminer une fonction, nous allons examiner sur quelles propriétés nous pouvons nous appuyer pour construire effectivement une fonction de reconnaissance.

4-2 PROPRIETES DE L'ESPACE DES REPRESENTATIONS INITIALES

En II nous avons défini une représentation initiale ou donnée initiale comme liste de valeurs.

$$X = (x_1, x_2, \dots, x_n)$$

Soit E l'ensemble des données initiales X . Chaque valeur X est le résultat d'une mesure et peut être interprétée. Il est important de préciser à quel type d'ensemble ou espace (ensemble doué de propriétés), appartiennent ces "valeurs".

Elles peuvent être *qualitatives*, exemple un nom parmi des noms, *quantitatives*, exemple une valeur numérique, rationnelle ou réelle.

Elles peuvent être *binaires*. Deux valeurs sont alors possibles. Binaires qualitatives si ces valeurs sont des noms, binaires quantitatives si ce sont des nombres binaires.

Nous noterons par des lettres *italiques* une variable X et par X l'espace des X . Il faut remarquer que l'identification de X à \mathbb{R}^n est un cas très particulier: alors toutes les valeurs x_i sont des réels et peuvent être additionnées et multipliées.

ORDRE SUR LES COUPLES, RESSEMBLANCE, DISSEMBLANCE

L'information donnée par l'observateur est un ordre partiel sur les couples.

$$(X_i, X_j) < (X_i, X_k) \quad (3)$$

On dit alors que X_i ressemble plus à X_i qu'à X_k , ou que X_k est plus dissemblant de X_i que X_j . L'implémentation pratique d'un tel concept intuitivement très important ne peut être obtenue que par une fonction constructive portant sur des couples de variables. Soit $\tau(X, Y)$ une fonction de ressemblance, $\mu(X, Y)$ de dissemblance. Ces fonctions sont à valeurs réelles, leur domaine de définition est en général $X \times X$.

$$\text{Si (3) est vérifié} \\ \tau(X_i, X_j) > \tau(X_i, X_k) \quad (\text{ressemblance})$$

$$\mu(X_i, X_j) < \mu(X_i, X_k) \quad (\text{dissemblance})$$

Il est clair qu'une infinité de τ ou μ induisent le même ordre sur les couples.

Un cas particulier d'une fonction de dissemblance est la *pseudo distance* satisfaisant aux propriétés axiomatiques

$$d(i, j) \leq d(i, k) + d(j, k)$$

$$d(i, i) = 0$$

ou la *distance* satisfaisant en plus à

$$d(i, j) = 0 \text{ entraîne } X_i = X_j$$

La donnée d'une distance entre représentations initiales est une donnée essentielle qui permet en général d'aboutir à une classification effective.

Nous allons voir comment de telles données peuvent permettre d'obtenir
- soit une meilleure description de l'ensemble des objets que la description initiale
- soit une identification.

ANALYSE EN COMPOSANTES PRINCIPALES

Cette technique appelée aussi analyse factorielle est particulièrement utilisée en Sciences Naturelles par les spécialistes de la "Taxinomie numérique". Les différents x_i sont des mesures numériques sur l'objet à étudier. Elles sont comparables entre elles. Il est donc légitime d'assimiler X à \mathbb{R}^n .

Les plans principaux d'inertie de l'ensemble E des données X_i sont calculés, grâce aux distances entre objets. Ces derniers sont projetés sur le premier plan et parfois aussi sur le second. Ces représentations permettent souvent de distinguer des classes disjointes et d'aboutir à une identification par la projection d'un X_i sur une région du plan. L'école de BENZECRI [2] a développé particulièrement ces techniques en France. Consulter aussi SNEATH et SOKAL [41].

ARBRE DE LONGUEUR MINIMUM

A partir de d , il est possible de construire sur E un "Arbre de Longueur Minimum" (A.L.M.). L'ALM est le graphe non orienté, connexe, sans circuit et de longueur totale minimum reliant tous les points de E .

Utilisant des heuristiques inspirées par le "bon sens", il est possible de séparer l'ALM en classes disjointes YOLKINA et ZAGORUJKO [49]. Pour cela on utilise un certain nombre de paramètres:

- h nombre de X_i dans une classe
- P proximité des X_i dans une classe
- λ égalité de distribution dans une classe
- d distance entre classes

et une fonction d'évaluation $F(h, P, \lambda, x)$

$$F = \text{Ln} \frac{d^a h^c}{(1+P^m)(1+\lambda^b)}$$

En général $a = c = m = b = 1$. Les classifications obtenues pour les plus grandes valeurs de F correspondent en général aux choix a priori de praticiens.

ETABLISSEMENT D'UNE HIERARCHIE

Le concept de hiérarchie est utilisé depuis fort longtemps par les "systématiciens" des Sciences Naturelles. Il permet d'introduire une "structure" dans un ensemble E d'objets. Cette structure est formée de sous ensembles de E , disjoints ou sinon emboîtés.

Le problème posé est d'obtenir une hiérarchie à partir des distances entre objets. La réponse n'est pas unique comme nous allons le voir.

L'algorithme habituellement utilisé est le suivant, LANCE et WILLIAMS [22]:

- Pas 1 Calculer la table triangulaire T des $d(i, j)$
- Pas 2 Soit (p, q) la paire telle que $d(p, q)$ est la plus petite distance de la table.
Alors supprimer q , remplacer p par r , recalculer les $d(i, r)$ pour tous les i restant dans T .
- Pas 3 Si le nombre de colonnes (ou de lignes) est plus grand que 1 aller au pas 2, sinon FIN.

On voit aisément que les p, q, r, i de l'algorithme ne sont rapidement plus des objets élémentaires mais des groupes d'objets. Il importe de définir la *distance d'un objet à un groupe* ou de *groupes entre eux*. Suivant cette définition le résultat final diffère.

Arrêter l'algorithme précédent avant la **FIN** permet d'obtenir des classes disjointes d'objets, donc une identification.

RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

PARTITIONS ADAPTATIVES

Nous allons maintenant nous intéresser aux techniques de partition adaptatives, dont l'idée est l'adéquation d'une partition en classes disjointes avec les représentations des classes.

Principes

Soient $P = P_1, P_2, \dots, P_k$ une partition de E en k classes disjointes.

Soient $A = A_1, A_2, \dots, A_k$, les "représentations" des k classes.

Soit $W(A, P)$ une fonction à valeur dans \mathbb{R}^+ exprimant l'adaptation réciproque de A à P . Nous la précisons par la suite. W peut être interprétée comme une ressemblance ou une dissemblance de A et P . Dans le premier cas, on cherche à la maximiser, dans le second à la minimiser.

Certains essais ont été faits, k étant variable, mais la plupart des efforts ont utilisé un nombre de classes k fixe.

Décrivons deux algorithmes types, dans lesquels A_i est simplement un objet de la classe P_i .

Donc $A_i = X_j \in P_i \subset E$.

Algorithme α

Pas 1 (initialisation)

Choisir k objets de E , comme "représentation" d'une classe.

Soient $A_1, \dots, A_j, \dots, A_k$.

Pas 2 (groupement)

$\forall i, X_i \in E$ appartient à P_j , ssi

$d(X_i, P_j) < d(X_i, P_e), \forall e \neq j$

Pas 3 (représentation adaptée)

$\forall j$ un A_j optimum est choisi pour P_j

Pas 4 Si aucun A_j n'est nouveau fin, sinon aller au Pas 2.

Algorithme β

Pas 1 le même que pour α

Pas 2 un élément X_i appartient à P_j , ssi

$d(X_i, P_j) < d(X_i, P_e), \forall e \neq j$.

Pas 3 un A_j optimum est choisi pour P_j

Pas 4 Si $\forall X_i \in E$ ont été placés dans une classe Fin, sinon aller au Pas 2.

Dans les deux algorithmes, on remarque que le classement des X_i (Pas 2) est fait par la technique des fonctions caractéristiques, cf.3-3.

Dans l'algorithme α , les représentations A_i ont changé quand tous les éléments X_i ont été affectés à une ou l'autre classe, cf. FORGEY [11] HALL et BALL [17], JANCEY [20].

Dans l'algorithme β , les représentations A_j peuvent varier après l'affectation d'un objet X_i à P_j . Ce dernier procédé est mieux adapté à l'arrivée séquentielle de nouvelles données, cf THORNDIKE [55], Mc QUEEN [56].

De nombreux perfectionnements ont été introduits, cf ANDERBERG [1], chapitre 7. Ainsi l'algorithme ISODATA de HALL et BALL n'introduit pas moins de sept seuils paramétrés pour contrôler le groupement. L'idée générale reste bien néanmoins d'adapter itérativement une description symbolique A des classes et une partition P .

N.B. Il se peut que certaines classes soient vides ou contiennent peu d'éléments à la fin du processus. Elles sont supprimées.

ALGORITHMES DES NOYEAUX DYNAMIQUES

Cet algorithme proposé par DIDAY [5] [7] originellement, généralise l'algorithme α . Une version "séquentielle", DIDAY [6], généralise l'algorithme β .

L'idée nouvelle est de mieux choisir les représentations A_i , qui étaient précédemment soit un élément de E , soit un point de X par exemple un centre de gravité.

Adaptons comme DIDAY le nom de "noyau" pour A_j .

L'idée de départ était de prendre pour noyau A_i les n_i premiers éléments de P_j tels qu'une fonction de dissemblance $R(X_i, P_j)$ soit minimum (Pas 3 de α). Une autre fonction de dissemblance $D(X_i, A_j)$ permet de classer un X_i dans une classe, comme précédemment (Pas 2 de α)

Soit $u_n = (A^{(n)}, P^{(n)})$ un couple de noyaux et partition, obtenus à une étape.

i) L'utilisation de R permet de passer de

$$(A^{(n)}, P^{(n)}) \text{ à } (A^{(n+1)}, P^{(n)}),$$

meilleure adaptation des noyaux à la partition $P^{(n)}$.

ii) L'utilisation de D , permet de classer l'ensemble E des points, c'est-à-dire d'obtenir une meilleure adaptation de la partition aux noyaux

$$(A^{(n+1)}, P^{(n)}) \rightarrow (A^{(n+1)}, P^{(n+1)})$$

L'utilisation alternée de R et D permet donc en deux étapes le passage de u_n à u_{n+1} .

L'initialisation de l'algorithme se fait en choisissant soit un A initial et en appliquant ii), soit un P initial et en appliquant i).

Le critère W est donné par:

$$W = \sum_{j=1}^k \sum_{X_i \in A_j} R(X_i, P_j)$$

Un choix quelconque de D et R ne permet pas d'assurer la décroissance de W . Mais en pratique, le plus souvent R et D sont déduits de $d(X, X')$, distance entre deux points de X par le même type de relation.

$$D(X, A_j) = \frac{1}{n_j} \sum_{X' \in A_j} d(X, X')$$

$$R(X, P_j) = \sum_{X' \in P_j} d(X, X')$$

$$W \text{ s'écrit alors } W = \sum_{j=1}^k W_j(P_j, A_j) \quad (4)$$

$$\text{avec } W_j = \sum_{X' \in A_j} \sum_{X \in P_j} d(X, X') \quad (5)$$



RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

Il est possible de démontrer la décroissance de W donnée par (4) et par (5), en même temps que progresse la suite u_n .

En pratique, étant donnée la précision finie des calculs, W arrive à une valeur W_m , borne inférieure de la suite $W^{(n)}$. Il se peut que la suite des couples u_n garde la même valeur ou passe de façon cyclique à travers un certain nombre de valeurs, pour lesquelles bien entendu $W = W_m$.

La valeur finale W_m dépend de la valeur initiale de u , donc du tirage de départ soit de A , soit de P . Il se peut que un ou plusieurs minimum partiels soient atteints. Ils correspondent à une ou des partitions "intéressantes" expérimentalement. Les situations stables correspondant au minimum minimorum de W ne sont pas forcément les plus "intéressantes".

En général, l'expérimentateur choisit des noyaux de départ "raisonnables" et, effectuant un certain nombre de choix différents, examine les différentes situations stables, choisissant parmi celles-ci, celles qui lui paraissent répondre le mieux à son problème de partition.

Différents types de noyaux ont été utilisés:

- un point, le centre de gravité ou le point de E le plus proche du centre de gravité
- un groupe de point
- une densité de probabilité
- un repère vectoriel, par exemple un plan, une droite.

D'une façon générale un noyau A_i peut être toute représentation adaptative de la classe P_i . Il importe

simplement d'être capable de définir

$D(X, A_i)$ distance d'un objet X au noyau A_i

$R(A_i, P_i)$ distance d'un noyau A_i à la classe P_i .

Remarques

1/ Rien n'assure que le minimum W^* atteint par W est le plus petit possible. Suivant l'initialisation de l'algorithme, différents W^* peuvent être atteints. A chacun d'eux correspond une partition P^* . L'intersection des P^* permet de définir le concept de "formes fortes". Cf SIMON, DIDAY [35]. Ces formes fortes permettent de donner une forme constructive aux ensembles flous (fuzzy sets) dont on parle beaucoup, mais qu'on ne voit pas souvent.

2/ Il a été proposé et utilisé par DIDAY et GOVAERT [8] des distances variables en même temps que les A_i et P_i . Bien entendu ces distances appartiennent à une classe déterminée de distance, par exemple les distances quadratiques. L'A.N.D est alors triangulaire.

4-3 CONSTRUCTION DE FONCTION DE RESSEMBLANCE OU DE DISTANCE.

Le choix, la mise au point des différentes fonctions de ressemblance ou de distance est fondamental, puisque à partir de cette donnée les différents algorithmes se déroulent de façon automatique.

Ces fonctions doivent exprimer les "propriétés" du problème.

Nous reportons à DIDAY [7] ou [8] pour une discussion du choix d'une distance entre objets. Le choix d'une "bonne" distance est souvent difficile, particulièrement quand les données sont qualitatives.

La ressemblance ou dissemblance d'un objet à un groupe d'objets ou à une classe sont fréquemment utilisées pour "classer" un objet: une fonction caractéristique n'est autre qu'une telle ressemblance ou dissemblance. Nous en avons déjà parlé quand nous avons décrit au § III la forme des fonctions de reconnaissance.

Enfin reste à considérer la ressemblance ou dissemblance d'une "représentation" à une classe d'objets, dont un exemple est le $R(A_i, P_i)$ de l'A.N.D. Cette question est encore plus délicate que les précédentes: comment peut-on estimer l'adéquation d'une représentation symbolique à une classe d'objets? Quelques exemples peuvent être trouvés dans DIDAY, SIMON [7] pp 74, 75, 79. Bien entendu une telle ressemblance doit traduire l'adéquation de la représentation symbolique (le noyau) à la classe d'objet.

Nous avons supposé que les objets X étaient en nombre fini et tous comparables, $X \in E$. L'ensemble E est en quelque sorte une discrétisation de l'espace topologique X . Ou si l'on préfère X est une extension de E . Savoir dans quelle mesure cette extension est justifiée et si les classes sont "séparables" par les techniques de fonctions caractéristiques est un problème qui ne peut avoir de réponse que domaine par domaine.

L'état actuel des résultats des recherches expérimentales montre clairement qu'il n'est pas possible de donner un diagnostic a priori. C'est aux résultats obtenus que l'expérimentateur juge si sa méthode était adaptée ou non au domaine envisagé.

Il est certain que l'existence de fonctions de ressemblance ou dissemblance "raisonnables", c'est-à-dire bien adaptées à la réalité, est une présomption de succès. Malheureusement, le praticien a tendance à appliquer telle ou telle formule de distance et telle ou telle technique, parce qu'il la connaît ou en a entendu du bien. La déception n'est alors pas loin...

Disons également que ces techniques sont tentantes pour un physicien. Les espaces envisagés sont voisins d'espaces de la physique ou de la mécanique. Des techniques variationnelles sont utilisées. Nous avons souligné les analogies avec les théories du potentiel ou de l'inertie, comparé à une énergie le critère W de l'algorithme des nuées dynamiques.

Il semble toutefois que ces méthodes doivent être abandonnées dès que l'on approche d'un niveau plus élevé de reconnaissance, qui correspond par exemple à la linguistique. Alors deviennent importantes les techniques de l'Intelligence Artificielle, auxquelles s'apparentent les méthodes évoquées au paragraphe 6. Mais malgré les efforts des spécialistes d'Intelligence Artificielle, ces méthodes n'ont guère conquis leur place en reconnaissance de la parole ou reconnaissance d'images, qui continuent à utiliser des techniques fondées sur la continuité de l'espace des représentations. Il en est de même dans l'analyse des signaux et l'analyse des données médicales, géologiques ou économiques.

En particulier, en économétrie, les spécialistes ne semblent pas pouvoir présenter d'autres sémantiques que des données de ressemblance, des distances. Les techniques de groupement y ont donc une importance spéciale.

§ 5 UTILISATION DE CONCEPTS D'INVARIANCE

5-1 Les idées d'invariance sont fondamentales en Mécanique et en Physique. En fait presque toutes les "lois" peuvent être déduites de principes variationnels, témoignant d'une forme d'invariance.

En Reconnaissance des Formes, reconnaître que des objets appartiennent à la même classe est leur attribuer une *invariance*. En fait reconnaissance et invariance sont alors des concepts très voisins. Tous les objets d'une classe possèdent la même propriété: "appartenir à la classe". Cette propriété est le résultat d'opérateurs de reconnaissance ou d'un opérateur d'identification, qui opèrent sur les données primitives. Nous avons déjà dit que la découverte d'une forme élémentaire (feature) est la découverte d'une invariance du problème.

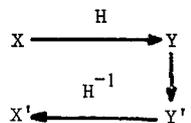
RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

La mise en évidence d'une invariance permet de réduire la *dimensionnalité* du problème. C'est bien évident si l'invariance est une reconnaissance: elle projette un objet X à n dimensions dans un ensemble fini.

Nous allons en voir d'autres aspects importants.

REDUCTION DE LA DIMENSIONALITE PAR INVARIANCE DE LA REPRESENTATION

Soit une représentation X d'un objet.
 $X \in \mathcal{X}$, espace de dimension n.
 Soit $Y = H(X)$ une transformation sur X,
 $Y \in \mathcal{Y}$ espace de dimension n. Pour fixer les idées, H est une transformation linéaire.
 Opérons sur Y en prenant q composantes parmi n (souvent les q premières). Soit Y' la nouvelle transformée. Revenons dans \mathcal{X} par H^{-1} .

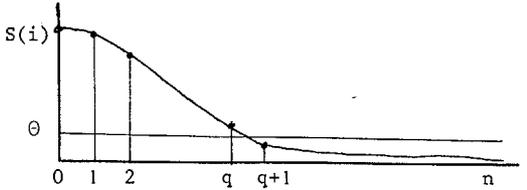


Il est fréquent qu'un observateur considère la représentation X' comme aussi "satisfaisante" que X. Dans ces conditions Y' est une représentation de dimension q, souvent très inférieure à n, utilisable pour poursuivre la reconnaissance, ou simplement plus économique que X.

- Comme exemple de H citons:
- La Transformée de Fourier à une ou deux dimensions
 - Les filtres digitaux linéaires (prétraitement des images)
 - Les Transformées de Karhunen-Loeve, Walsh-Hadamard
 - L'Analyse de données.

Les applications sont importantes en Traitement des Images. Elles sont souvent tentées en traitement de signal et de parole.

La justification physique d'un tel processus est en général la suivante:



Soit $S(i)$ le "spectre" de l'espace \mathcal{X} c'est-à-dire le graphe des n valeurs moyennes quadratiques entrées de Y. En général $S(i)$ a l'allure de la figure ci-contre. La valeur de q est fixée par un seuil θ . On dit qu'en-dessous de θ les fluctuations des composantes d'ordre élevé correspondent à un "bruit". Un tel point de vue, en général valable aux niveaux les plus bas de la reconnaissance, est parfois tout à fait injustifié. Il se peut que l'une des composantes de rang élevé soit au contraire tout à fait "signifiante". cf. FUKUNAGA [14], ROSENFELD [32], HUANG [19].

Il est important de comprendre la portée pratique de tels procédés généraux. Apparemment il est séduisant de réduire la dimensionnalité d'un problème par un calcul général d'aspect "savant". On aurait ainsi des recettes générales qui pourraient toujours s'appliquer. En réalité il faut les employer avec discernement et bien peser les conséquences en temps de calcul et volume de mémoire. Cependant pour certaines transformations telles que la Transformation de Fourier Rapide, il existe des microcircuits cablés très performants.

4-4 INVARIANCE AUX TRANSFORMATIONS.

Soit une représentation X d'un objet et T une transformation sur l'objet tel que TX soit la nouvelle représentation. On souhaite que l'opération d'identification donne le même résultat sur X et sur TX. Une manière évidente de procéder est évidemment que $X = TX$, c'est-à-dire que la représentation soit invariante aux transformations T.

Une telle idée est fréquemment utilisée, donnons en quelques exemples.

RECONNAISSANCE DE LA PAROLE

L'emploi de spectrogrammes revient à utiliser une représentation invariante en fonction de transformations sur le signal de parole, par exemple déphasage des composantes fréquentielles en fonction de la propagation.

Certains reconnaisseurs codent ces spectrogrammes par 0 et 1 en fonction du signe de la dérivée, représentation invariante à l'amplitude. En fait seuls sont représentés la position des extrema. du spectre, en particulier des "formants", maxima du spectre. La position des formants est reconnue comme particulièrement insensible aux transformations du signal dues à la propagation, aux variations d'amplitude.

RECONNAISSANCE DES IMAGES

En reconnaissance de caractères, la représentation d'une lettre est ramenée à un format déterminé de façon à tenir compte de diverses transformations. D'autre part l'utilisation de formes caractéristiques (features) telles que bâtons, points de rebroussement, de croisement provient clairement du fait qu'ils ont une forme invariante en fonction de nombreuses transformations ou déformations.

Citons certaines représentations invariantes de nuages de points. SIMON [36].

Soit une figure de points $A_1 \dots A_i \dots A_n$. Si nous ordonnons les segments formés par les points $A_i A_j$, pris deux à deux, la suite ordonnée des couples est invariante en fonction d'une rotation, d'une homothétie. Bien d'autres représentations peuvent être proposées. Soit un des segments $A_i A_j$, il sépare le plan en deux parties, donc l'ensemble des points restants en deux sous-ensembles, qui d'ailleurs peuvent être ordonnés. Cette donnée est invariante en fonction d'une projection plane.

§ 6 CONSTRUCTIONS HEURISTIQUES D'OPERATEURS DE RECONNAISSANCE

6-1 Dans cette dernière partie, nous abordons les contributions les plus originales de la Reconnaissance des Formes.

Nous avons déjà souligné que, de même qu'il importe de ne pas séparer représentation et interprétation, il importe de ne pas séparer une forme (feature) de son opérateur d'identification. Une forme "idéale" telle qu'une droite, un cercle, est en fait définie par l'homme qui en "voyant" une représentation est capable de décider que c'est une droite, un cercle.

La Reconnaissance des Formes se trouve donc dans la situation de connaître les deux extrémités de la chaîne et de devoir la reconstituer par des opérateurs réels construisibles, plus précisément par des programmes.

Un problème d'Intelligence Artificielle peut se mettre sous une forme semblable:



RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

On part d'une situation initiale, décrite par certaines propriétés. L'utilisation d'un certain nombre d'opérateurs permet de modifier la situation. Le but cherché est d'aboutir à une situation finale déterminée. Il est hors de question de chercher à faire une combinaison exhaustive des opérateurs disponibles. On cherche à guider la construction par des "heuristiques", qui traduisent en fait le comportement humain en résolution de problèmes. PITRAT [29], LAURIERE [23].

Le problème de la Reconnaissance des Formes est plus simple en ce sens que la situation finale est une identification mais aussi plus complexe parce que nous ignorons pratiquement tout des processus physiologiques et psychologiques qui nous permettent de reconnaître des formes.

Nous allons voir comment ces idées sont appliquées concrètement.

6-2 SYSTEMES INTERACTIFS DE RECONNAISSANCE DES FORMES

Un programme de Reconnaissance des Formes plus particulièrement d'identification, s'écrirait comme un programme algorithmique à partir d'instructions ou d'opérateurs élémentaires.

Par exemple, le "prétraitement" de données fournies par un capteur se ramène à l'emploi d'un certain nombre d'opérateurs tels que:

- extraction des maximum
- recherche de gradients
- recherche des éléments dépassant un seuil dans une liste ou un tableau
- filtrage, etc...

Habituellement le programmeur guidé par son intuition et son expérience va combiner les opérateurs dont il dispose en machine. Il va en inventer d'autres, adaptés à son problème, cf par exemple le filtre de formants proposé par J.C. SIMON et B. DUPEYRAT [40].

Il est utile d'évaluer les performances de tels opérateurs et de leurs combinaisons. C'est pourquoi différents systèmes *interactifs* de Reconnaissance des Formes ont été proposés et construits.

Citons en exemple le système OLPARS du ROME AIR FORCE DEVELOPMENT CENTER (U.S.A.).

Citons en France le système développé par C. ROCHE [30] à l'Etablissement Central d'Armement. Il est également décrit par FERTE [10]. Les données sont entrées sous formes de scalaires, de listes, ou de tableaux. Les différents opérateurs disponibles (les instructions du langage) sont par exemple:

- Somme et moyenne des éléments d'une liste, d'un tableau.
- Exponentiation d'une liste, d'un tableau.
- Produit ou quotient d'une liste, d'un tableau par un scalaire.
- Opérateur somme/différence/produit/quotient de deux scalaires d'une liste/d'un tableau.
- Opérateur gradient horizontal ou vertical pondéré sur un tableau.
- Opérateur lissage d'une liste.
- Position, valeur, largeur au sommet et nombre de maximum d'une liste.
- Opérateur seuil sur scalaire/liste/tableau.
- Opérateur affectation (rangement en mémoire).
- Opérateurs logiques (résultats vrai ou faux).
- Opérateur histogramme d'une liste/tableau.
- Opérateur conditionnel (permettant la récursion).
- Opérateur DO (permettant l'itération).

On note immédiatement que sont introduites non seulement des opérations *algébriques* utilisant les opérateurs arithmétiques usuels mais aussi des opérations *logiques* et *conditionnelles*, en fait la puissance de calcul des *fonctions récursives* au sens des informaticiens.

UTILISATION D'OPERATEURS SYNTAXIQUES

Dans le même esprit, beaucoup de reconnaisseurs ont découvert l'emploi de *règles de réécriture*, couramment appelées opérations *syntaxiques*. cf par exemple FU [12], FU et ROSENFELD [13] p 1339. En fait de telles opérations syntaxiques sont "plus élémentaires" en Informatique que les opérations arithmétiques. Il est clair que le retard à les employer provient du bagage culturel des reconnaisseurs et aussi du fait que les ordinateurs ont d'abord été conçus comme des machines arithmétiques.

Citons l'emploi systématique de règles de grammaires pour différents problèmes pratiques tels que la reconnaissance de la parole HATTON [18], ROCHE [31], des empreintes digitales MOAYER et FU [25], de l'interprétation de photos de satellite FU [54], etc...

Rappelons toutefois que l'intérêt essentiel de l'emploi de grammaires formelles est leur caractère *récuratif*, sinon rien ne distingue une telle technique d'une simple recherche en mémoire. Une grammaire récursive permet par contre de réaliser une économie de mémoire qui peut être considérable.

Le problème de l'inférence d'une grammaire formelle à partir d'exemples et de contre-exemples est un problème souvent mal posé. Il a intéressé de nombreux chercheurs, cf par exemple dans J.C. SIMON [39] Part 3.

Citons enfin l'emploi de grammaires stochastiques et de grammaires d'images.

COMBINAISONS DE TESTS, QUESTIONNAIRES

Il est fréquent que l'expérimentateur soit amené à proposer un ensemble de tests t_i , c'est-à-dire d'opérateurs dont le résultat est "Vrai" ou "Faux". Comment construire un opérateur d'identification correct à partir de ces tests? L'opérateur en question est souvent appelé un *questionnaire*. Ce sujet a été abordé dès 1963 par PICARD [28]. Signalons les travaux de TERRENOIRE [43]. Cette école de chercheurs tente d'aboutir à une identification correcte en utilisant le plus petit nombre de tests ou les tests les moins "coûteux".

GAILLAT [57] s'est posé le problème d'utiliser un ensemble de tests séparant des objets deux à deux.

PROCESSUS ADAPTATIFS

Il arrive que certains reconnaisseurs décident a priori d'utiliser une fonction paramétrée déterminée; par exemple l'opération linéaire $H.X$, H étant un vecteur de même dimension que X . Ils cherchent ensuite à adapter les paramètres, ici les composantes de H , au problème posé.

Cette technique ressemble aux procédés d'interpolation ou d'extrapolation de fonctions à partir d'échantillons et de polynômes.

Partant de l'ensemble d'apprentissage T , des procédés itératifs peuvent permettre d'adapter H au problème. Cette technique était fort à la mode, il y a une quinzaine d'années, sous le terme de PERCEPTRON. NILSSON [26] MINSKI et PAPPERT [24] en ont montré les limites.

Il est clair que, si un tel opérateur peut être adapté à un ensemble d'apprentissage suffisamment petit, rien ne garantit ses qualités de *généralisation*. Pour fixer les idées, utilisant l'exemple, rien ne garantit que deux classes pourront être séparées par un hyperplan. cf DUDA et HART [9] Chapitre 5 pour de nombreux exemples d'opérateurs adaptatifs.



RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

6-3 UTILISATION D'HEURISTIQUES GÉNÉRALES, FONCTIONS D'ÉVALUATION

Il semble essentiel de chercher à évaluer les performances d'un opérateur ou de la combinaison de plusieurs opérateurs. C'est l'idée d'heuristiques de l'Intelligence Artificielle. On peut se demander quel est le bien fondé d'heuristiques générales. Elles peuvent être justifiées si on est capable d'établir une statistique des performances de fonctionnement.

Soit $X \in \mathcal{X}$ une donnée. Soit $\omega(i)$ une forme à reconnaître. Soit \mathcal{E}^* l'opérateur idéal dont le résultat sur X sera un $\omega(i)$. Soit \mathcal{E}_j un opérateur expérimental dont on veut évaluer les performances. Soit $a_j(k) \in A_j$ les réponses de A_j . Par des essais, il est possible d'établir des valeurs approchées des probabilités

$$p[\omega(i), a_j(k)]$$

probabilité d'apparition simultanée de $\omega(i)$ et $a_j(k)$, en présentant différents X .

De même si on connaît $p(\omega_i)$ et $p(a_j)$

$$p[a_j(k) | \omega(i)], p[\omega(i) | a_j(k)]$$

par le théorème de Bayes. Par exemple

$$p[\omega(i) | a_j(k)] = \frac{p[\omega(i), a_j(k)]}{\sum_i p[\omega(i)] p[a_j(k) | \omega(i)]}$$

A partir de ces données, différentes fonctions d'évaluation permettent d'estimer les différences de performances entre l'opérateur expérimental \mathcal{E}_j et l'opérateur idéal \mathcal{E}^* .

La fonction la plus souvent utilisée est l'*information mutuelle* au sens de Shannon entre \mathcal{E}_j et \mathcal{E}^* .

$$I[A_j; \Omega] = \sum_{i,k} p[\omega(i), a_j(k)] \log \frac{p[a_j(k) | \omega(i)]}{p[a_j(k)]}$$

Si $I[A_j; \Omega] = H[\Omega]$, l'opérateur \mathcal{E}_j est idéal

Si $I[A_j; \Omega] = 0$, l'opérateur \mathcal{E}_j n'a pas d'intérêt

Parmi les choix possibles, on fera celui pour lequel la fonction I est maximum. La quantité $H-I$ est analogue à une distance à l'opérateur idéal. On montre d'ailleurs que si la *probabilité d'erreur* P_e est petite, $H-I \cong P_e \log P_e$.

Ces concepts introduits par BONGARD et LEWIS ont été étudiés de façon approfondie et appliqués par ROCHE [31], voir aussi SIMON et ROCHE [37], SABAH [33] [34].

Ils ont cherché non seulement à évaluer la performance d'un opérateur, mais aussi à le construire sous la forme d'une table optimale. Également à orienter le choix des données d'entrée. Les résultats semblent souvent intéressants, sans que le succès soit toujours assuré. C'est un sujet de recherche encore très ouvert.

D'autres fonctions d'évaluation ont été proposées comme l'*entropie quadratique*, DEVIJVER [4].

Pour une liste assez complète voir KANAL [52].

Il est à remarquer que dans la littérature américaine ce sujet est appelé "feature evaluation", alors qu'il apparaît que sont évalués des opérateurs et non des formes, mais il est vrai que nous avons souligné qu'on ne saurait séparer l'un de l'autre...

Ces techniques permettent d'obtenir un opérateur d'identification, dont les performances sont satisfaisantes sur un ensemble T d'apprentissage. Mais rien ne permet de supposer que les qualités de *généralisation* de l'opérateur sont satisfaisantes en dehors de l'ensemble T . Les déceptions sont parfois graves.

D'autre part il est nécessaire de connaître suffisamment bien $p(\omega, a)$, donc d'avoir une connaissance statistique des performances de l'opérateur A_j . Ce dernier point est particulièrement critiquable. Faut-il pour "apprendre" avoir une connaissance statistique du problème? Ne peut-on plutôt apprendre en généralisant un petit nombre d'exemples. Cf SIMON [39].

Ces idées ont entraîné tout un ensemble de nouvelles recherches évoquées dans ce qui suit.

6-4 SYSTEMES DE QUESTION REPONSE ET APPRENTISSAGE

Le problème de compréhension d'un langage naturel est aussi un problème de reconnaissance. Une question devant entraîner une réponse, le problème peut se ramener à établir l'opérateur qui, à partir d'une question en entrée, donnera une réponse correcte. Dans une certaine mesure, ces opérateurs généralisent les opérateurs de reconnaissance.

Ce point de vue s'est avéré fructueux:

La question en langage naturel est d'abord analysée non seulement par la syntaxe mais aussi par la sémantique et traduite dans une structure profonde SABAH [58]. A ce niveau se posent les problèmes d'organisation en mémoire et de représentation des connaissances. En particulier de nombreux chercheurs tels que SHANK, CHARNIAK, BOBROW, WINOGRAD ont étudié comment représenter et utiliser des connaissances usuelles, des "know how".

Il est probable que ces études d'Intelligence Artificielle auront une répercussion importante sur la manière de concevoir et de construire les grandes banques de données spécialisées, dont on souhaiterait qu'elles réagissent comme un être humain compétent et intelligent.

En particulier de tels systèmes de Question-Réponse sont capables d'absorber de façon intelligente des connaissances nouvelles, c'est-à-dire de *réorganiser* leurs bases de données. C'est en ce sens qu'on peut dire qu'ils "*apprennent*". Ces recherches s'appuient sur les résultats des recherches récentes des linguistes et des sémanticiens. Il est clair que le système possède a priori beaucoup de capacités assez complexes, représentées par des opérateurs ou programmes eux-mêmes très élaborés. Il est hors de question (pour le moment) de trouver des heuristiques capables de synthétiser les programmes eux-mêmes, comme il était proposé au paragraphe précédent.

INFERENCE DE FONCTIONS RECURSIVES

D'autres recherches s'attachent à résoudre un problème voisin; trouver un opérateur récursif capable de généraliser une transformation "Question-Réponse": A partir d'un ensemble d'exemples $\{[X_i \rightarrow F(X_i)]\}$ (l'ensemble d'apprentissage T), trouver une fonction F *récursive*, fonctionnant de façon satisfaisante non seulement sur les exemples donnés, mais aussi sur d'autres exemples fournis par le "professeur". Il va sans dire que ceci suppose d'une part que le professeur applique une "loi générale" et d'autre part ne change pas de loi au cours de ses exemples.

SUMMERS [42] et JOUANNAUD, GUIHO [21] viennent d'obtenir des résultats très intéressants dans cette ligne de recherche. Pour fixer les idées donnons quelques exemples.



RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

Question X	Réponse F(X)	Fonction F
A B C D	B C D A	le premier devient le dernier
A B C D	D C B A	retourner la liste
A B C D	ABCDCBA	miroir
A B C D	AABCCDD	doubler
A B C D	AABABCBCD	répétition itérative
A B C D E F	A F E D C B	renverser une liste partielle
A B C D E F	A F B E C D	mélanger des cartes de jeu
ABCDEFHIJK	KBIDGFEHCJA	idem
A B C D E F	A B C	liste moitié
A B C D	ABBCCDDDD	répéter suivant le rang
A B C D E F	C B A F E D	renverser les listes moitié

En particulier le système SISP de JOUANNAUD est capable, à partir d'un seul couple d'exemples, de trouver une fonction récursive F satisfaisante, sous la forme d'un programme LISP, avec condition d'arrêt.

SISP n'a aucune information sémantique ou linguistique. La technique employée est en gros de décomposer les deux listes grâce à des fonctions élémentaires et par des heuristiques adaptées. Puis de découvrir dans le graphe joignant X et F(X) des chemins mettant en évidence une fonction récursive possible.

En fait les connaissances "a priori" du système sont beaucoup moins complexes que précédemment, mais le langage est très restreint et les capacités très spécialisées.

Il semble toutefois que ces recherches jettent un jour très nouveau sur les problèmes de la découverte d'opérateurs à partir d'exemples par des techniques *non statistiques* et sur les capacités de généralisation.

De même qu'en Physique les équations différentielles et aux dérivées partielles permettent une économie de représentation considérable, les fonctions récursives permettent de représenter des lois, parfois complexes, avec grande économie. Leur importance doit être soulignée en Reconnaissance des Formes et Traitement de Signal.

BIBLIOGRAPHIE

- [1] M.A. ANDERBERG "Cluster analysis for applications", Acad. Press N.Y. (1973).
- [2] J.P. BENZECRI L'analyse de données, Tome I et II (Dunod, Paris 1973).
- [3] T.M. COVER "A hierarchy of probability density function estimates" in Frontiers of Pattern Recognition. S. Watanabe edit. (Acad. Press N.Y.) (1972).
- [4] P.A. DEVIJVER Entropie quadratique et R.F. dans J.C. Simon, Computer oriented learning processes Noordhoff Int. Leyden [1976].
- [5] E. DIDAY "Une nouvelle méthode de classification automatique et reconnaissance des formes: la méthode des nuées dynamiques" Revue de Statistique appliquée. T. 19, n°2 (1970).
- [6] E. DIDAY "The dynamic clusters method and sequentialization in non hierarchical clustering" Proc. First Int. Joint. Conf. Pattern Recognition. IEEE Special Publi. n°73, CH 0821-9C (1974).
- [7] E. DIDAY, J.C. SIMON Clustering analysis, dans K.S. Fu Digital Pattern Recognition, Springer Verlag Berlin, Heidelberg 1976.
- [8] E. DIDAY, G. GOVAERT Adaptive resemblance measure in P.R. dans J.C. Simon Computer oriented learning processes p. 229 Noordhoff Int. Leyden 1976.
- [9] R.O. DUDA, P.E. HART Pattern classification and scene analysis Wiley, N.Y. (1971).
- [10] J.F. FERTE Recherche automatique des paramètres caractéristiques en R.F. Thèse de 3è cycle, juin 1976. Document ECA 76-R-118, Arcueil.
- [11] E.W. FORGEY Cluster analysis of Multivariate data A.R.A.S. Biometric Society (W.N.R.R.) Riverside, Calif. USA (1965).
- [12] K.S. FU Syntactic methods in P.R. and machine learning, N.Y. Acad. Press 1968.
- [13] K.S. FU, A. ROSENFELD Pattern recognition and image processing IEEE Transaction on computers. C-25, n°12 Dec. 1976.
- [14] K. FUKUNAGA, W. KOONTZ "Application of the Karhunen-Loève expansion to feature selection and ordering" IEEE Trans. Comput. vol. C-19, pp.311-318. April 1970.
- [15] K. FUKUNAGA Introduction to statistical pattern recognition. N.Y., Acad. Press, 1972.
- [16] J. GIPS Shape grammar and their uses, Birkhauser Verlag, Bâle 1975.
- [17] HALL et BALL A clustering technique for summarizing multivariate data Behavioral Science, vol.12, n°2 (1967).
- [18] J.P. HATTON, J.M. PIERREL Reconnaissance automatique de la parole, p.101. Actes du Congrès AFCET Paris 3-5 nov. 1976.
- [19] T.S. HUANG Picture processing and digital filtering, N.Y. Springer 1975.
- [20] R.C. JANNEY "Multidimensional group analysis" Aust. Jour. Bot. vol.14, p.127 (1966).
- [21] J.P. JOUANNAUD, G. GUIHO Inference of function with an interactive system dans Michie edit. Machine Intelligence 9. Edimburg 1977.
- [22] G.N. LANCE, W.T. WILLIAMS A general theory of classification sorting strategy Computer Journal, vol.9, pp 1973 (1967).
- [23] J.L. LAURIERE Un langage et un programme pour les problèmes combinatoires p. 257. Actes du Congrès AFCET. Paris, 3-5 nov. 1976.
- [24] M. MINSKY, S. PAPPERT Perceptrons, and introduction to computational geometry The MIT Press, Cambridge USA (1969).
- [25] B. MOAYER et K.S. FU A tree system approach for fingerprint pattern recognition IEEE Trans. Comput. vol C-25 mar. 1976.
- [26] J.N. NILSSON Learning machines, foundation of trainable pattern classifying systems (Mc Graw Hill, Hill, N.Y. 1965).

RECONNAISSANCE DES FORMES ET TRAITEMENT DES IMAGES

- [27] E.A. PATRICK Fundamentals of Pattern Recognition Englewood Cliffs NJ, Prentice Hall 1972.
- [28] C.F. PICARD Graphes et Questionnaires, Gauthier-Villars (1972).
- [29] J. PITRAT Directions actuelles de la recherche en A.I. p.243. Actes du Congrès AFCET. Paris, 3-5 nov. 1976.
- [30] C. ROCHE Generation of recognition operators by stochastic and heuristic learning, dans J.C. Simon Computer oriented learning processes, Noordhoff Int. Leyden (1976).
- [31] C. ROCHE Thèse d'Etat, Univ.Paris VI, déc. 1972.
- [32] A. ROSENFELD Digital Picture analysis, New York, Springer 1976.
- [33] G. SABAH Thèse de 3^è cycle, I.P.Paris VI, nov. 1973.
- [34] G. SABAH The notion of command operator in pattern recognition. Proceedings Second I.J.C.P.R. p.376, Copenhagen 1974.
- [35] J.C. SIMON, E. DIDAY Classification automatique C.R. Acad. Sc. de Paris, t. 275, (13 nov.1972) Série A.
- [36] J.C. SIMON, A. CHECROUN et C. ROCHE A method of comparing two patterns independent of possible transformations and small distortions. Pattern Recognition 4, 73 (1972).
- [37] J.C. SIMON et C. ROCHE On automatic generation of P.R. operators, Proceedings de IEEE Int. Conf. on Cybernetics and Society p. 232 (1972).
- [38] J.C. SIMON Recent progress to a formal approach of pattern recognition and scene analysis. Pattern Recognition Journal vol. 7 pp.117-124 (1975).
- [39] J.C. SIMON Computer oriented learning processes, Noordhoff Int., Leyden, 1976.
- [40] J.C. SIMON et B. DUPEYRAT Analyse rapide des composantes formantiques d'un signal de parole. C.R. Ac. Sc. Séance du 28 fév. 1977.
- [41] P. SNEATH et R. SOKAL Numerical Taxonomy W.M. Freeman edit. (1974).
- [42] P.D. SUMMERS A methodology for LISP program construction from examples. J.A.C.M. 24 1977.
- [43] M. TERRENOIRE Etudes de diverses heuristiques pour des processus d'interrogation dans J.C. Simon. Computer oriented learning processes, Noordhoff Int. Leyden 1976.
- [44] J.T. TOU et R.C. GONZALES. Pattern Recognition principles. New York, Addison Wesley, 1974.
- [45] J.P. TREUIL, G. GUIHO, J.P. JOUANNAUD Une méthode d'apprentissage de concepts p.77 Congrès AFCET, Paris, 3-5 nov. 1976.
- [46] L. UHR Pattern Recognition, learning and thought. Prentice Hall, 1973.
- [47] J.K. ULLMANN Pattern Recognition technique, Crane, Russak and Co 1973.
- [48] S. WATANABE Frontiers of Pattern Recognition, New York, Academic Press, 1972.
- [49] VN. YOLKINA et NG ZAGORUIKO Quantitative criteria of Taxinomy Quality and use in the process of decision making; Vychislitel'nye Systemy, 36, Computer Systems, Institute of mathematics, Novosibirsk (en russe).
- [50] F. de SAUSSURE Cours de linguistique générale, Payot, Paris, 1972.
- [51] Z. MANNA Mathematical theory of computation Mc Graw Hill, N.Y. 1974.
- [52] L. KANAL Patterns in Pattern Recognition. IEEE Transactions on Information Theory, vol. IT-20, n°6, nov. 74.
- [53] G.C. STOCKMAN, L. KANAL. An experimental waveform parsing system. Proceedings of the 2nd I.J.C.P.R. Copenhagen, Aug.74.
- [54] K.S. FU Digital Pattern Recognition. Springer Verlag, Berlin, 1976.
- [55] R.L. THORNDIKE Who belongs in the family. Psychometrika, vol. 18, pp.267-276 (1953).
- [56] J. Mc QUEEN Some methods for classification and Analysis of multivariate observations 5th Berkeley Symposium on Mathematics, Statistic and Probability, 1967.
- [57] G. GAILLAT Une procédure statistique de décision avec apprentissage... Thèse de 3^è cycle, mai 1975, Univ. Paris VI.
- [58] G. SABAH Un système d'apprentissage et de compréhension du langage naturel, p. 269, Actes du Congrès AFCET, Paris, nov. 1976.
- Proceedings Int. Joint Conf. on Pattern Recognition *Première*, Washington D.C., Oct.1973. *Seconde*, Copenhagen, Dk, Août 1974. *Troisième*, Coronado, Cal. USA, nov. 1976

