

Module - Apprentissage (PRTAL 5)

D.E.A Informatique, 2000-2001

9 janvier 2001

1 Catégorisation d'images

1.1 Définition du problème

Une image est une matrice m_{ij} de $n \times n$ de pixels, ayant chacun une valeur réelle entre 0 (noir) et 1 (blanc). On obtient une image à partir d'une image en niveaux de gris en choisissant une zone $n \times n$ quelconque de l'image. On désire mettre en œuvre un procédé d'auto-organisation connexionniste afin de catégoriser les images prises aléatoirement au sein d'une image en niveaux de gris. On souhaite « mettre dans le même paquet » les images qu'un humain jugerait similaires « à l'œil ».

Questions préliminaires :

1. Décrivez les différentes étapes d'un mécanisme d'auto-organisation connexionniste, ces étapes étant communes aux mécanismes que vous connaissez.
2. Précisez le rôle de la notion de distance, dans la ou les étapes dans lesquelles cette notion intervient.
3. Parmi les procédés d'auto-organisation connexionnistes que vous connaissez, lequel préconisez vous ici ? (justifiez votre réponse)

Le but du problème, et donc des questions suivantes, est de trouver une distance adaptée à la catégorisation d'images.

1.2 Comparaison d'images

1.2.1 La distance euclidienne

Pour comparer deux images m et m' de taille $n \times n$, on se propose d'utiliser la distance euclidienne définie par l'équation 1.

$$d(m, m') = \sqrt{\sum_{i=1}^n \sum_{j=1}^n (m_{ij} - m'_{ij})^2} \quad (1)$$

Afin de juger de la pertinence du choix de la distance euclidienne, répondez aux questions suivantes

1. Pour les deux couples (m, m') de la figure 1, donnez la distance euclidienne entre les deux images. Si on utilise cette distance, comment se fera la catégorisation ?
2. En quoi la forme mathématique de la distance euclidienne fait qu'elle est mal adaptée à notre problème ?

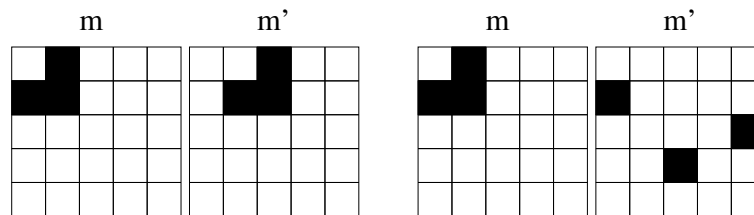


FIG. 1 – Deux couple d'images à comparer

1.2.2 Une distance plus adaptée

La notion de distance, au sens strict du terme, n'est pas nécessaire. Il suffit de mettre en œuvre un procédé pour qu'un neurone réagisse fortement pour une image m' proche de son image m (la « valeur nominale » du neurone), et qu'il ne réagisse pas si on lui présente une image m' « différente » de m . **On rappelle que la notion d'image proche doit correspondre le mieux possible au jugement de ressemblance que ferait un humain.**

1. Proposez une fonction de réponse du neurone d'image prototype m , qui n'ait pas le défaut de la distance euclidienne. Cette fonction retournera une valeur d'autant plus proche de 1 que l'image m' présentée « ressemble » à m , et une valeur proche de 0 si m et m' ne se ressemblent pas.
2. Proposez une règle d'apprentissage pour rapprocher le prototype m de l'entrée m' .

Dans les deux cas, on ne vous demande pas LA solution parfaite, mais de proposer une idée, d'en présenter les points forts et éventuellement les limites.

2 Prédateur et proie

2.1 Définition du problème

On se pose le problème du contrôle d'un robot prédateur (cf. figure 2), qui doit attraper une proie pour la consommer (il peut y avoir plusieurs proies, mais on supposera qu'à un moment donné, **une seule est dans le carré perceptif**). Les proies décrivent une trajectoire quelconque (vitesse et direction variable), et le prédateur perçoit une proie lorsqu'elle se situe dans un carré centré autour de lui¹, comme l'illustre la figure 2. Le monde se comporte comme suit. Le temps est discret, et à chaque pas de temps, le robot perçoit l'état de chacune des 49 cases de son carré perceptif. L'état d'une case est binaire : présence ou absence du centre de la proie. Le robot dispose de deux roues latérales, qu'il peut tourner

¹Le carré perceptif tourne et se déplace avec le prédateur.

indépendamment vers l'avant de 4 valeurs discrètes, de 0 à \max pas. Par exemple, 0 sur la roue droite et une des 3 autres valeurs sur la roue gauche provoque une rotation vers la droite, autour de la roue droite. Une fois l'action effectuée, le robot en perçoit le résultat au pas suivant, la proie ayant éventuellement bougé dans le monde. Le but du robot est qu'un maximum de proies arrivent en case centrale de son carré perceptif (le robot est sur la proie), auquel cas les proies disparaissent (le prédateur les a automatiquement « mangées »).

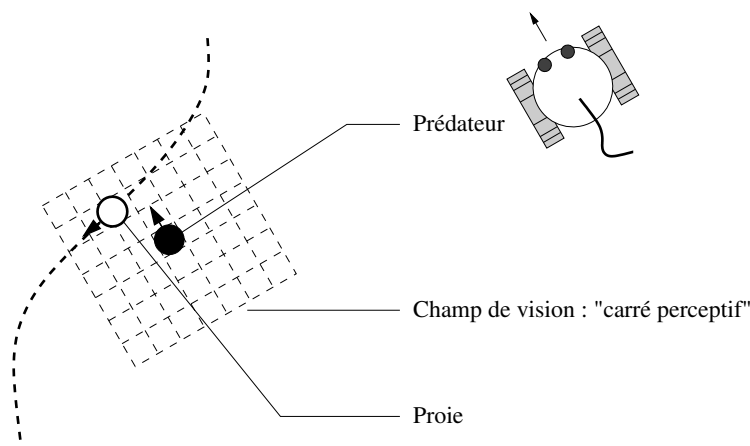


FIG. 2 – Prédateur et proie.

On souhaite résoudre le problème par apprentissage par renforcement, en utilisant un processus de décision markovien (MDP).

1. Sachant qu'il n'y a toujours qu'une seule proie visible dans le champ de perception, quel est le nombre d'états possibles du carré perceptif? Quel est le nombre d'actions possibles du robot?
2. Pourquoi ne peut-on pas utiliser les états perceptifs du robot comme états du MDP?
3. Que se passerait-il si malgré tout on le faisait?

2.2 Construction du MDP

On suppose maintenant que les proies ont une vitesse constante (différente d'une proie à l'autre), et une trajectoire rectiligne. On conserve l'hypothèse qu'une seule proie est visible à un moment donné.

1. Proposez un ensemble d'états qui convienne pour la résolution à l'aide d'un MDP.
2. Combien de Q -values cet ensemble d'état occasionne-t-il?
3. Quelle fonction de récompense proposez-vous pour tenir directement compte de l'objectif du prédateur?
4. Comment adapter la fonction de récompense pour éviter que la proie sorte du champ de vision.

2.3 Résolution du MDP

1. Décrivez la mise en œuvre d'une méthode de Monte Carlo pour résoudre le MDP précédent.
2. Décrivez la mise en œuvre d'un algorithme de *Q-Learning*.