

PLANÈTE ROBOTS

NOUVELLES TECHNOLOGIES DU FUR

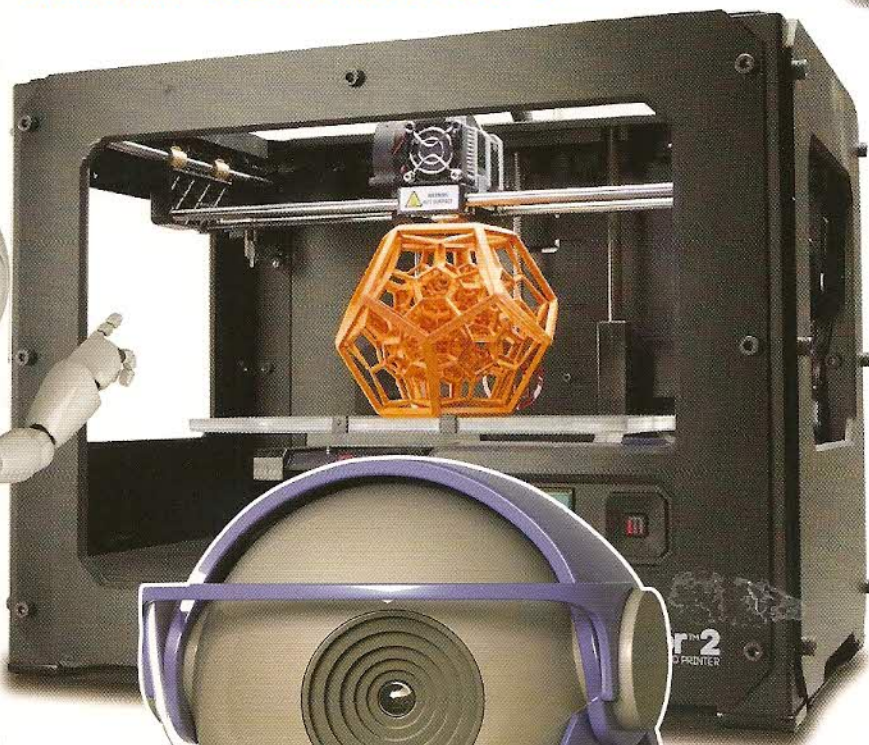


**LES
ROBOTS
ANIMAUX
DU CHIEN
AU GUÉPARD**

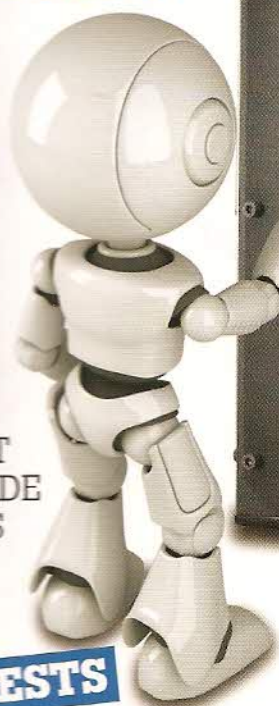
RÉVOLUTION ! IMPRIMANTES 3D

FABRIQUE TOI-MÊME TON ROBOT !

**LES FAMEUX
FAB LABS**



**ARIA,
STAR DE
LA BNF
UN ROBOT
HUMANOÏDE
FRANÇAIS**

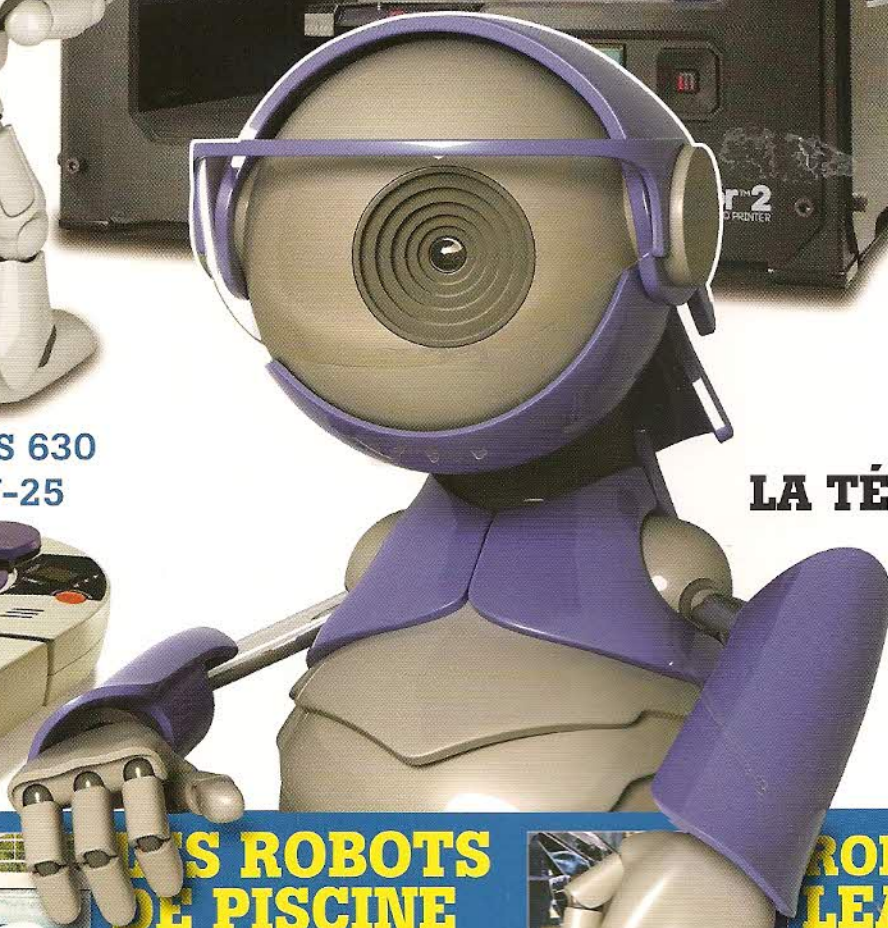


NOS TESTS

- ROBOMOW RS 630
- LE NEATO XV-25



**LA MAGIE DE
LA TÉLÉPORTATION
QUANTIQUE
L'ÉTAT DE LA
RECHERCHE**



**LES ROBOTS
DE PISCINE**

**NOTRE SÉLECTION
DE 12 MODÈLES**



**ROBOT COMBAT
LEAGUE**

**DES ROBOTS S'AFFRONTENT
DEVANT LES CAMÉRAS**

UNE APPLICATION EN NEUROMIMÉTIQUE : LA RECONNAISSANCE DE LETTRES


Dans le numéro dix-neuf de *Planète Robots*, nous vous avons expliqué la particularité adaptative des réseaux neuromimétiques. Nous vous proposons ici une application des réseaux de neurones — en fait une approche particulière de l'informatique que l'on appelle *Neural Computing*. Ce domaine s'essaie à reproduire le fonctionnement du cerveau humain pour mieux le comprendre ou pour développer des applications déductibles liées à un dilemme.

Neural Computing - TAN - Toy Adaptive Node

Identification Forme

Coloriez les cellules pour représenter un modèle de lettre ou de forme

Lettre "T"

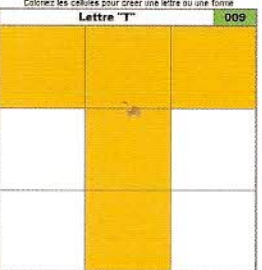


Demande d'identification

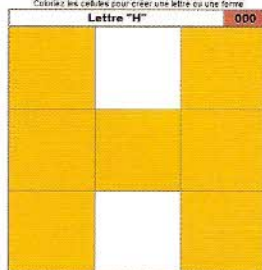
Modèles Formes

Coloriez les cellules pour créer une lettre ou une forme

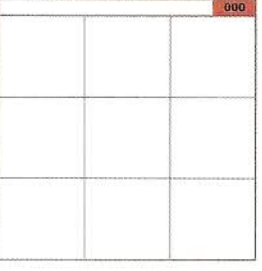
Lettre "T" 009



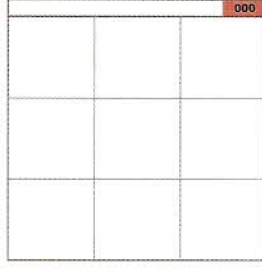
Lettre "H" 000



Coloriez les cellules pour créer une lettre ou une forme 000



Coloriez les cellules pour créer une lettre ou une forme 000



Configuration

Positionnement

Taille Forme

Hauteur: 03 blocs de 05 cellules.

Largeur: 03 blocs de 05 cellules.

Position Relative A La Référence

Nom de la forme: 01 lignes, 00 colonnes.

Score d'identification: 01 lignes, 15 colonnes.

Messages

Forme en recherche: Forme en recherche :0{

Forme non trouvée: Forme non trouvée :0{

Formes proches: Formes proches :0{

Couleurs

De changement:

Références Des Formes

L'ongle est en haut à gauche de la forme

Référence Identification Forme

SC38

Références Modèles Formes

SW58

SAP58

Image n°1

La reconnaissance de formes se prête bien aux réseaux neuronaux même si une autre approche utilisant l'informatique statistique se révèle envisageable. Le programme proposé est réalisé en langage VBA sous Microsoft Excel, vous pourrez l'utiliser et le modifier à loisir en le téléchargeant à cette adresse (<http://www.eveillenvol.com/planeterobots/planeterobots.htm>).

Il est basé sur les travaux d'Igor Aleksander et d'Helen Morton, les auteurs du livre *An Introduction to Neural Computing*.

LES DÉTAILS DE L'ALGORITHME DU PROGRAMME

Le principe de l'application consiste à apprendre les lettres « T » et « H » (le programme permet d'apprendre d'autres lettres ou des formes) dans une matrice

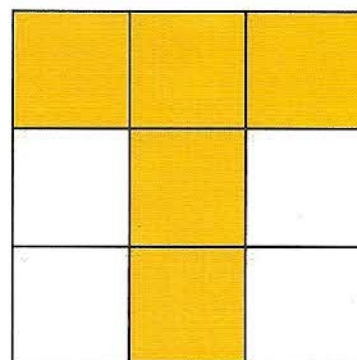


Image n°3

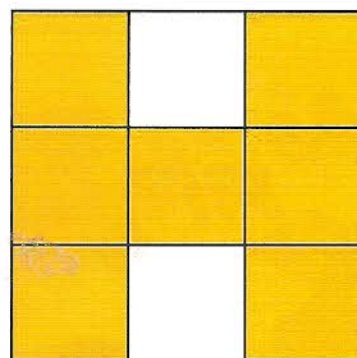


Image n°4

de trois x trois neurones qui représentent les pixels d'un affichage de lettres, des neuropixels (images n°3 et n°4)...

Lors de la présentation d'une lettre incomplète au réseau de neurones, il va s'auto-organiser et essayer de tendre vers une lettre apprise.

Dans l'exemple ci-dessous, on observe les cas possibles...

- N°5 C'est sûrement un T.
- N°6 C'est sûrement un H.
- N°7 Je ne sais pas...

Pour s'auto-organiser, chaque neurone — appelé aussi *Toy Adaptive Node* (TAN) — est relié à son voisin (image n°8)...

Pour obtenir le bon câblage et afin de gérer ses propres informations, le TAN se voit attribuer quatre entrées et une sortie. (Chaque entrée est reliée à la sortie de son voisin.) Le neurone possède aussi un

```

Microsoft Visual Basic pour Applications - ERégis-Project_PM_eveilEnvolNeuralComputingTAN_V1.0_FR.xls - [MMTAN (Code)]
L11, Col1
Fichier Edition Affichage Insertion Format Débogage Exécution Outils Compléments Fenêtre ?
Tapez une question
Projet - VBAProject
VBAPProject (ERégis-Project_PM)
  Microsoft Excel Objets
  Feuil3 (TAN)
  ThisWorkbook
  Feuilles
  LPStartBox
  Modules
  MMTAN
Propriétés - MMTAN
MMTAN Module
Alphabétique | Par catégorie
[blanc] MMTAN
(Général) (Déclarations)
' Auteur modificateur: Emmanuel REGIS, 14/09/2013.
' Le Toy Adaptive Node ou TAN, permet de simuler un neurone. Comme tout neurone, il y a une période d'apprentissage
' en fonction de ses entrées (ou modèle) et de sa sortie (ou état). Une fois l'apprentissage réalisé, si on lui
' propose un modèle, le TAN essaiera de l'interpréter et son état changera. Il faut remarquer que le TAN est à
' la base un composant électronique. De plus, il doit être connecté à d'autres TAN si on veut un réseau de neurones.
' Ce réseau diluera l'information pour mieux interpréter une information qu'il lui est inconnu.
' Dans ce programme le TAN représente l'environnement d'un TAN pixel.
' En entrée:  Mode du TAN, apprentissage ou non (VL_LearnOnOff).
'            Ligne et colonne du TAN pixel (VL_FormRowBlock, VL_FormColumnBlock).
'            Les informations du TAN pixel à mémoriser ou à évaluer (VL_TANPixel).
' En sortie:  Pour le mode restitution (Cf. Sans apprentissage), le retour de fonction renvoie:
'            1/ Une valeur nulle indique que le TAN n'a pas pu reconnaître le TAN pixel et donc aucun modèle de forme.
'            2/ Une valeur positive indique que le TAN a pu reconnaître le TAN pixel et donc un modèle d'une forme.
'            La valeur est la position du modèle de la forme dans la base de données.
Private Function MM_NeuralComputing_ComponentTAN(ByRef VL_LearnOnOff As Boolean, ByRef VL_FormRowBlock As Integer, ByRef VL_FormColumnB
Dim VL_CompareFormModelHandle As Integer
Dim VL_CompareTANPixelCounter As Integer
Dim VL_CompareTANPixelNumber As Integer
Dim VL_FormModelCounter As Integer
' Initialisation.
VL_CompareFormModelHandle = CE_NULL
' Mode apprentissage
If VL_LearnOnOff = CE_FLAG_OK Then
' Le TAN mémorise/apprend les informations du TAN pixel courant.
VL_FormModelValue(VL_FormModelNumber).TANPixel(VL_FormRowBlock, VL_FormColumnBlock) = VL_TANPixel
Else
' Mise à zéro du nombre maximal de paramètres identiques liés aux TAN pixels comparés.
VL_CompareTANPixelNumber = CE_NULL
' Le TAN recherche le TAN pixel au plus proche dans la base de connaissance des modèles de formes.
For VL_FormModelCounter = 1 To VL_FormModelNumber
With VL_FormModelValue(VL_FormModelCounter).TANPixel(VL_FormRowBlock, VL_FormColumnBlock)
' Mise à zéro du nombre courant de paramètres identiques liés aux TAN pixels comparés.

```

Image n°2

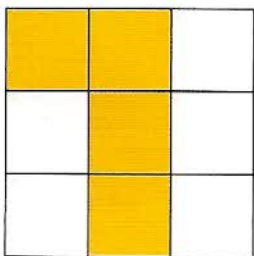


Image n°5

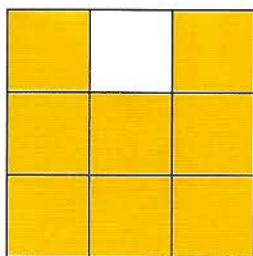


Image n°6

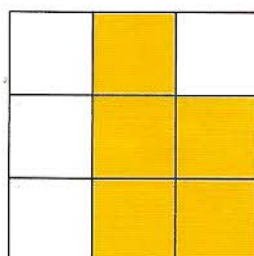


Image n°7

commutateur pour savoir s'il est en mode Apprentissage ou en mode Restitution. Les lignes des entrées acheminent des données au neurone puis, en fonction, il émet une information à la sortie. (Rappelons au passage que les sorties de la ma-

trice de trois x trois neurones représentent les états allumés ou éteints des pixels des lettres: les neuropixels.)

Le commutateur permet deux cas... — En mode Apprentissage, les données des entrées et de l'état de la sortie sont apprises. Ce mode est particulier à l'information en sortie qui à ce moment ne sort pas du neurone mais y entre.

— En mode Restitution, les entrées informent le neurone sur son environnement et il réagit en fournissant une information via sa sortie.

Les informations sur les lignes sont de type binaire — c'est-à-dire du courant sur la ligne ou non (cf. Marche / Arrêt, 0 / 1 ou encore Allumé / Éteint).

Ces neurones sont donc comparables à des composants électroniques qui peuvent être câblés à loisir. Mais à la différence de l'informatique classique, où une donnée est posée précisément dans une mémoire (cf. Adresse Mémoire), le câblage de nos neuropixels diffuse « dilue » l'information. Ainsi, les lettres

« T » et « H » ne sont donc pas précisément placées dans des mémoires — mais diluées dans le réseau de neurones.

Les neuropixels sont nommés dans notre cas par leur position Colonne / Ligne. TAN C1L1 indique par exemple un neurone en position Colonne N°1 et Ligne N°1 de la matrice (image n°9).

Une fois le câblage réalisé, le réseau est mis en apprentissage. Puis on présente chaque lettre une par une, en positionnant les sorties de chaque neurone. Dans notre réseau, on remarque qu'il est inutile de positionner les entrées car les sorties sont branchées aux entrées!...

Puis le réseau est mis en restitution et on présente une lettre incomplète. À ce moment, le réseau se réorganise. Chaque neurone voit sa sortie se modifier pour tendre vers un état qu'il a appris tout en tenant compte des sorties de ses voisins. Cette nouvelle configuration forme un nouvel affichage de la lettre recherchée grâce à nos neuropixels. L'utilisateur peut, s'il le souhaite, présenter à nouveau la

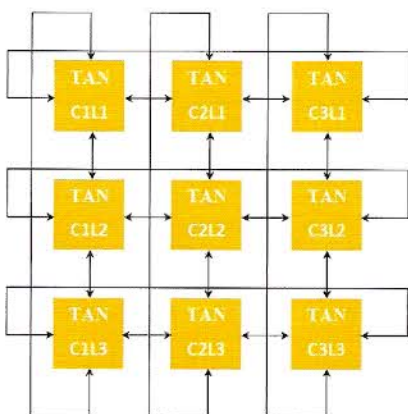


Image n°8



Image n°9

| | | |
|----|----|----|
| C1 | C2 | C3 |
| L1 | L1 | L1 |
| C1 | C2 | C3 |
| L2 | L2 | L2 |
| C1 | C2 | C3 |
| L3 | L3 | L3 |



| | | |
|---------------|---|---|
| TAN C1L1 | T | H |
| C3L1 (Gauche) | 1 | 1 |
| C1L3 (Haut) | 0 | 1 |
| C2L1 (Droite) | 1 | 0 |
| C1L2 (Bas) | 0 | 1 |
| Sortie | 1 | 1 |

| | | |
|----|----|----|
| C1 | C2 | C3 |
| L1 | L1 | L1 |
| C1 | C2 | C3 |
| L2 | L2 | L2 |
| C1 | C2 | C3 |
| L3 | L3 | L3 |

| | | |
|----|----|----|
| C1 | C2 | C3 |
| L1 | L1 | L1 |
| C1 | C2 | C3 |
| L2 | L2 | L2 |
| C1 | C2 | C3 |
| L3 | L3 | L3 |

Image n°13

Images n° 14 et 16

Un score est alors calculé par rapport aux entrées présentées et à celles qui ont été apprises pour chaque lettre. Et ici, le nombre d'entrées identiques donne la victoire à la lettre « T » (image n°12)... Ainsi, la sortie s'activera et le neuropixel s'allumera (image n°13)...

| | | |
|----|----|----|
| C1 | C2 | C3 |
| L1 | L1 | L1 |
| C1 | C2 | C3 |
| L2 | L2 | L2 |
| C1 | C2 | C3 |
| L3 | L3 | L3 |



Image n°10

| | | |
|----|----|----|
| C1 | C2 | C3 |
| L1 | L1 | L1 |
| C1 | C2 | C3 |
| L2 | L2 | L2 |
| C1 | C2 | C3 |
| L3 | L3 | L3 |

Image n°11

nouvelle organisation. Après divers essais et diverses itérations en chaîne, le réseau se stabilise vers la meilleure reconnaissance possible.

Voyons maintenant comment fonctionne précisément l'intérieur d'un neuropixel...

Pendant l'apprentissage (ici sous la forme d'un tableau), chaque neurone mémorise ses entrées et sa sortie. (Sachant que « 1 » dans une case signifie que la ligne a eu du courant ou que c'est allumé et que « 0 » indique qu'il n'y a pas de courant ou que c'est éteint...

| Apprentissage | | | Restitution | | | |
|---------------|---|---|-------------|--|---|---|
| TAN C2L3 | T | H | Comparaison | TAN C2L3 | Nombre d'entrées identiques pour le « T » : | Nombre d'entrées identiques pour le « H » : |
| C1L3 (Gauche) | 0 | 1 | | 0 (Gauche) | 4 | 1 |
| C2L2 (Haut) | 1 | 1 | | 1 (Haut) | | |
| C3L3 (Droite) | 0 | 1 | | 0 (Droite) | | |
| C2L1 (Bas) | 1 | 0 | | 1 (Bas) | | |
| Sortie | 1 | 0 | 0 ⇒ 1 | La sortie s'activera car le score du « T » est le plus élevé | | |

Image n°12

Voici par exemple les entrailles du TAN C1L1 (image n°10)...

Lors de la restitution, pour faire reconnaître une lettre incomplète, chaque neurone va comparer ses entrées à celles qui ont été apprises. Et les entrées présentées les plus proches de celles qui ont été apprises sélectionneront la nouvelle information de sortie.

Si l'on présente un « T » incomplet, voyons la réaction du TAN C2L3 (image n°11)...

En revanche, la sortie n'est pas modifiée si la comparaison est identique. On l'observe sur le TAN C2L2 avec cette nouvelle présentation d'une lettre incomplète (image n°14)...

Les entrailles du TAN C2L2 se mettent donc en action (image n°15)...

Le neuropixel en déduit que le nombre d'entrées identiques est le même quel que soit le modèle d'apprentissage; la sortie est indéterminée et reste alors inchangée (image n°16)...

Nous sommes arrivés au terme de cette application. Il existe en fait plusieurs types de structures de réseaux de neurones... Le nôtre est un réseau feed-back (car il reboucle). Il permet de reconnaître des lettres et peut être aussi dérivé pour reconnaître des situations, des risques, des stratégies et pour concevoir un jeu, pour construire un Tamagotchi et bien d'autres choses! À vous d'essayer, de dériver le programme Microsoft Excel et d'explorer d'autres applications ou d'autres types de réseaux ou de neurones...

| Apprentissage | | | Restitution | | | |
|---------------|---|---|-------------|---|---|---|
| TAN C2L2 | T | H | Comparaison | TAN C2L2 | Nombre d'entrées identiques pour le « T » : | Nombre d'entrées identiques pour le « H » : |
| C1L2 (Gauche) | 0 | 1 | | 0 (Gauche) | 2 | 2 |
| C2L1 (Haut) | 1 | 0 | | 1 (Haut) | | |
| C3L2 (Droite) | 0 | 1 | | 1 (Droite) | | |
| C2L3 (Bas) | 1 | 0 | | 0 (Bas) | | |
| Sortie | 1 | 1 | 0 ⇒ 0 | La sortie restera inactive car les scores sont identiques | | |

Image n°15