

# **RECONNAISSANCE DE FORMES PAR RESEAU DE NEURONES**

Timothée COUR

Guillaume GIRAUD

Antoine KODSI

Tuan-Anh LUONG

Rémy LAURANSON

Clémentine MARCOVICI

Kolia SADEGHI

## PRESENTATION ORALE

# PLAN

Introduction

Présentation du programme

Démonstration

Résultats

Présentation du travail en groupe

# DEFINITION

réseau de neurones  
**= réseau de neurones artificiels**

-> constitué de “neurones”

-> reliés par des **connexions**

-> capacité d'**apprentissage** = apprendre à partir d'exemples

=> généraliser, classer

# MODELISATION D'UN NEURONE

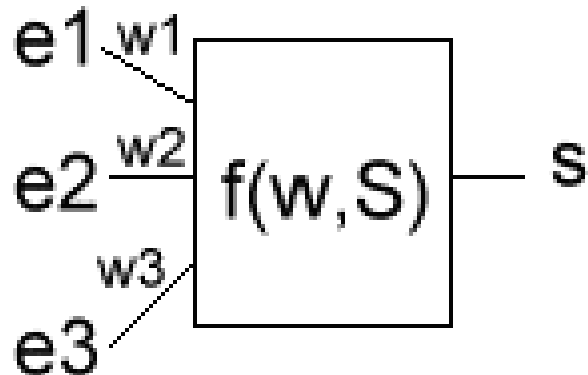


schéma d'un neurone artificiel

**entrées** :  $e1$ ,  $e2$ ,  $e3$

**sortie** :  $S$

**poids** :  $w1$ ,  $w2$ ,  $w3$

**fonction d'activation** :  $f$

**entrées** : e1, e2, e3

**sortie** : S

**connexion**

= canal de communication entre deux neurones (= **synapse**)

**poids** : w1, w2, w3

= coefficients numériques indiquant l'importance de la connexion

**fonction d'activation** f

= fonction de traitement des entrées fournissant la sortie du neurone

entrées (e1, e2, e3) + poids (w1, w2, w3)

$$E = w1 * e1 + \overset{\sim}{w2} * e2 + w3 * e3$$

||  
V

sortie S = f(E)

# MODELISATION D'UN RESEAU DE NEURONES

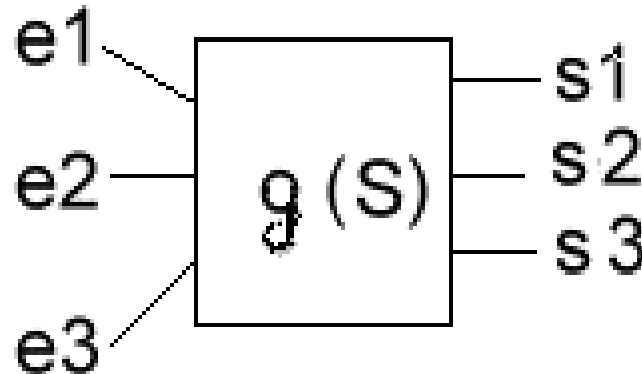


schéma général d'un réseau de neurone

**entrées** :  $e_1$ ,  $e_2$ ,  $e_3$

**sorties** :  $s_1$ ,  $s_2$ ,  $s_3$

poids variables -> problème = comment modifier les poids  
(phase d'apprentissage)

# FONCTIONS A REALISER PAR UN RESEAU DE NEURONES

Classification = reconnaissance **supervisée**

- apprentissage supervisé : on fournit les sorties désirées lors de l'apprentissage -> ajustement des poids

(perceptron multicouches)

Généralisation (trouver des ressemblances parmi des formes) = reconnaissance **non supervisée**

- apprentissage non supervisé : évolution libre -> stabilisation

(réseau de Kohonen)

# RECONNAISSANCE MANUSCRITE

Erreur humaine ~ 1/1000

Logiciel : pas encore satisfaisant

Réseau de neurones = outil privilégié

## Exemples d'application industrielle :

### - lecture automatique à la Poste

reconnaissance des adresses

1% confusion

### - lecture de chèques par la société A2iA

montant chiffre + lettre

1‰ confusion



# NOTRE PROJET

## **Problème posé :**

écrire et tester un programme de reconnaissance de formes implémentant un réseau de neurones.

Proche de la biologie ? originalité

Réseau de neurones classiques ? résultats concrets

-> « prendre le meilleur des deux »

Appliqué à la reconnaissance de chiffres manuscrits

## **Objectifs :**

atteindre un taux de reconnaissance supérieur à 75% sur la base de données NIST.

# Résultats :

- un programme abouti entièrement écrit par le groupe (6000 lignes de code)
- structure générale d'un Perceptron Multicouche à 7 couches (= aires)
- apport personnel : certaines couches originales -> proche de la biologie
- taux de reconnaissance ~ 80-90 %  
avec contrainte : faible nombre d'exemples donnés en apprentissage

# ALGORITHME GENERAL ET PROGRAMME INFORMATIQUE

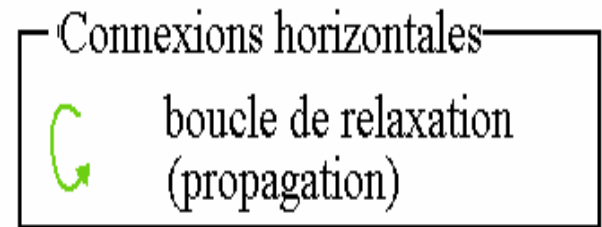
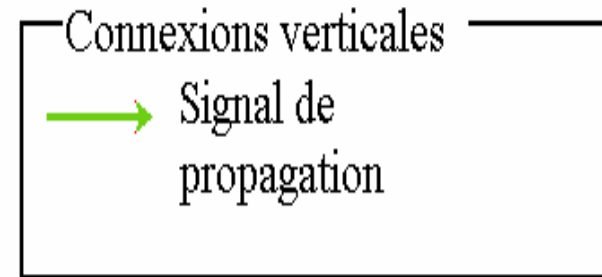
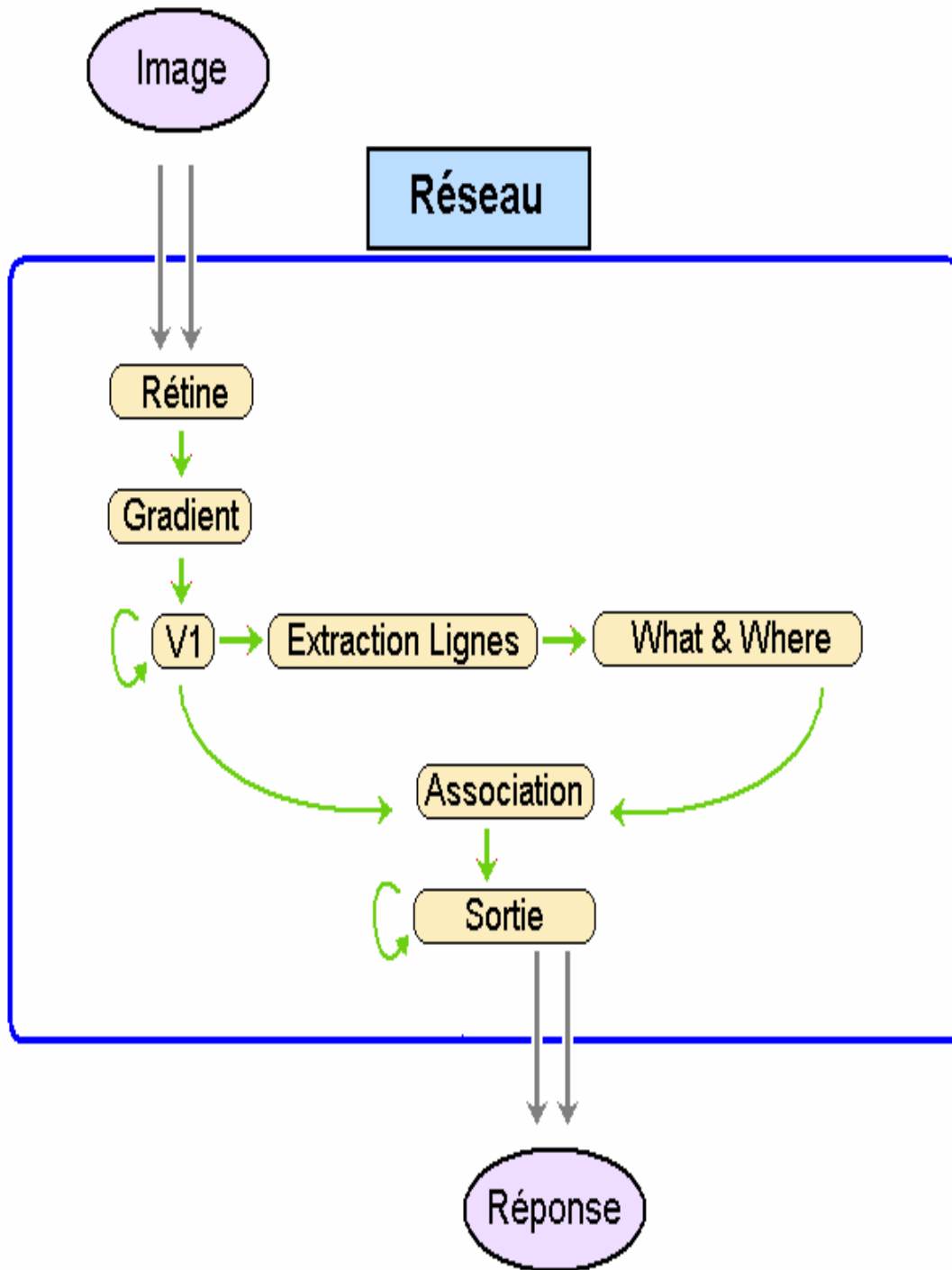
- Principes pour la conception du réseau de neurones
- Organisation du réseau en différentes aires
- Exemple d'aire : l'aire V1
- Le modèle de neurone utilisé
- Spécificité de notre approche

# Principes pour la conception du réseau de neurones

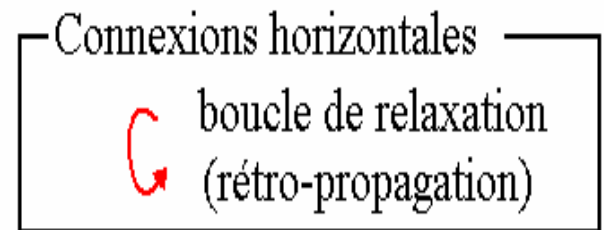
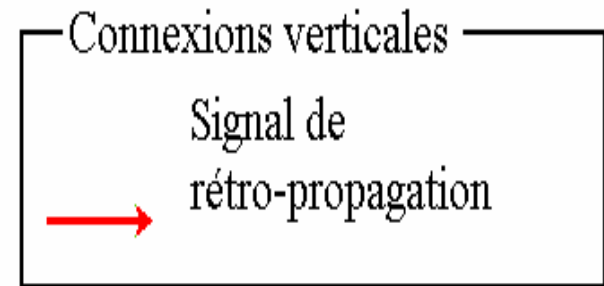
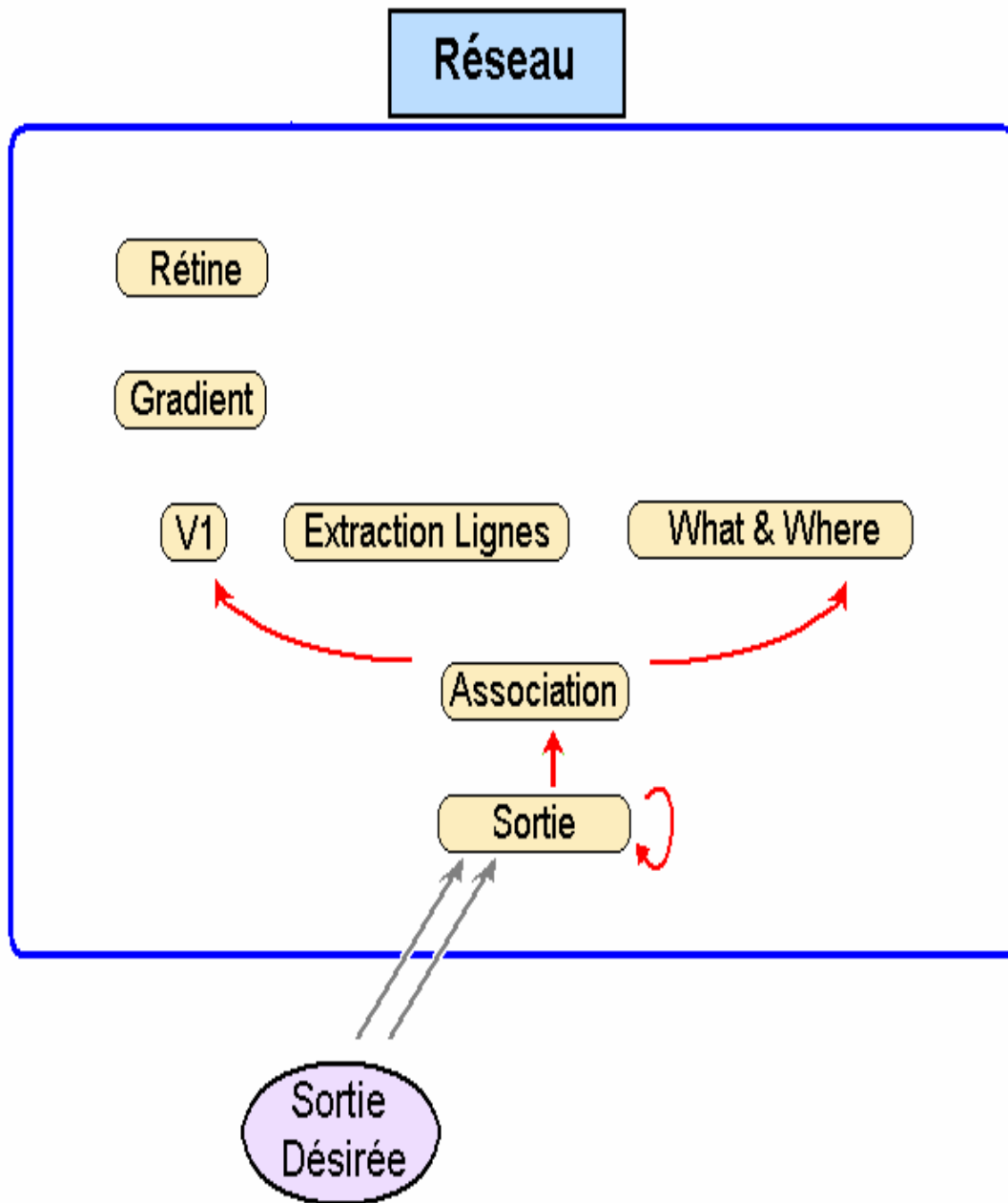
- Unité du système (prétraitements neuronaux)
- Réseau facilement reconfigurable
- Découpage fonctionnel du réseau en aires = couches
- Opérations uniquement locales

# Organisation du réseau en différentes aires

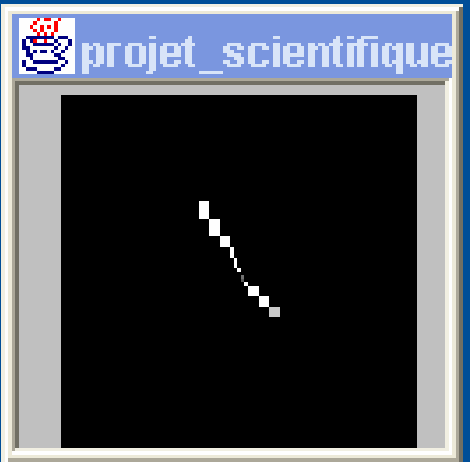
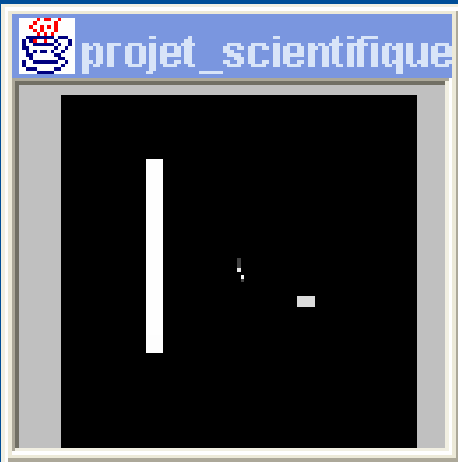
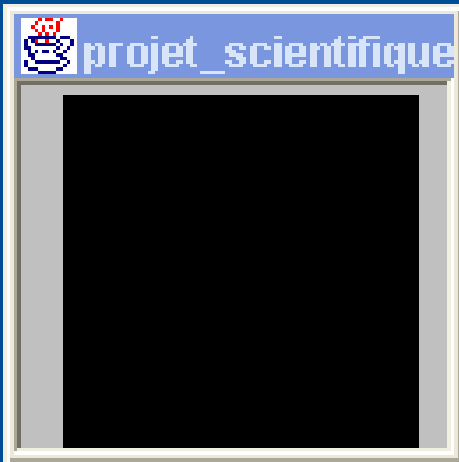
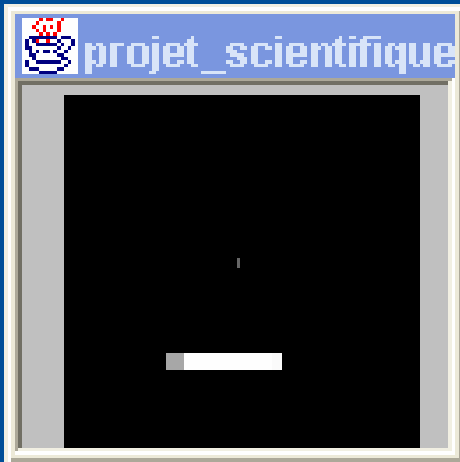
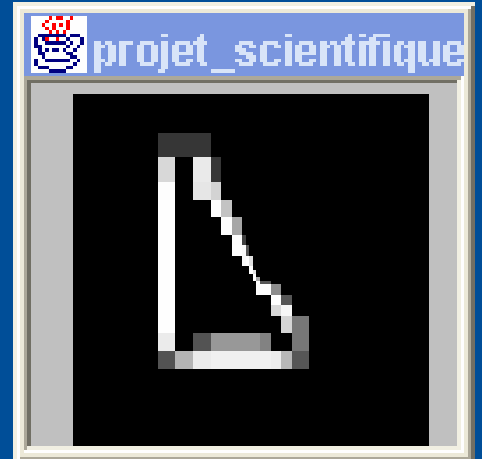
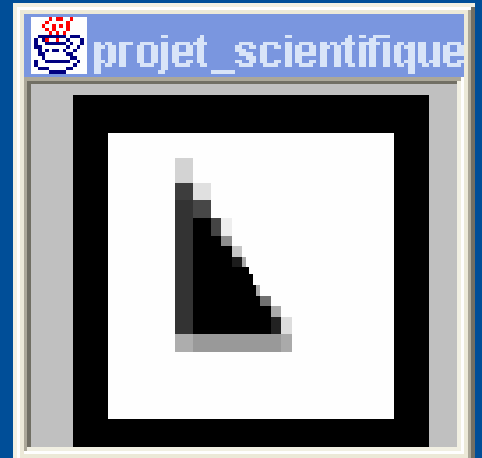
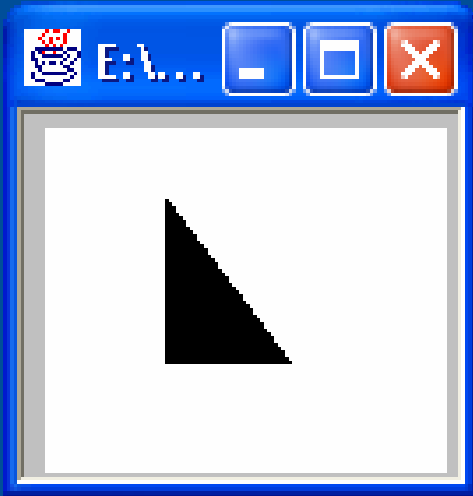
- Rétine : recevoir l'image
- Gradient : contraste
- V1 : micro-lignes
- Extraction de lignes,  
What&Where : extraire et positionner formes dans l'image
- Sortie : réponse du réseau



→ Informations "physiques" entre le réseau et l'environnement



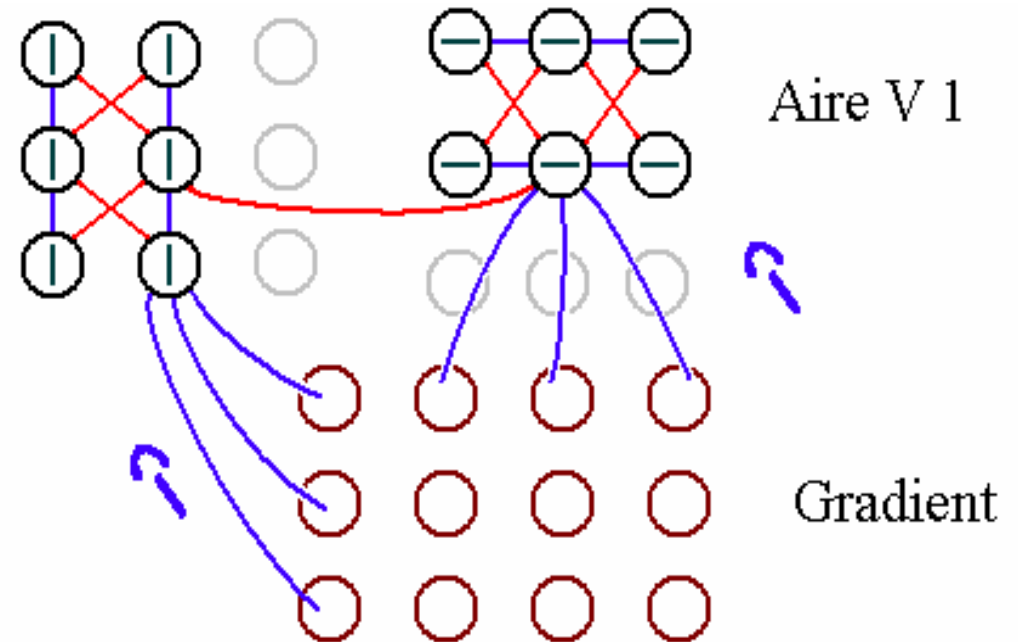
— Informations "physiques" entre le réseau et l'environnement





# Exemple d'aire : l'aire V1

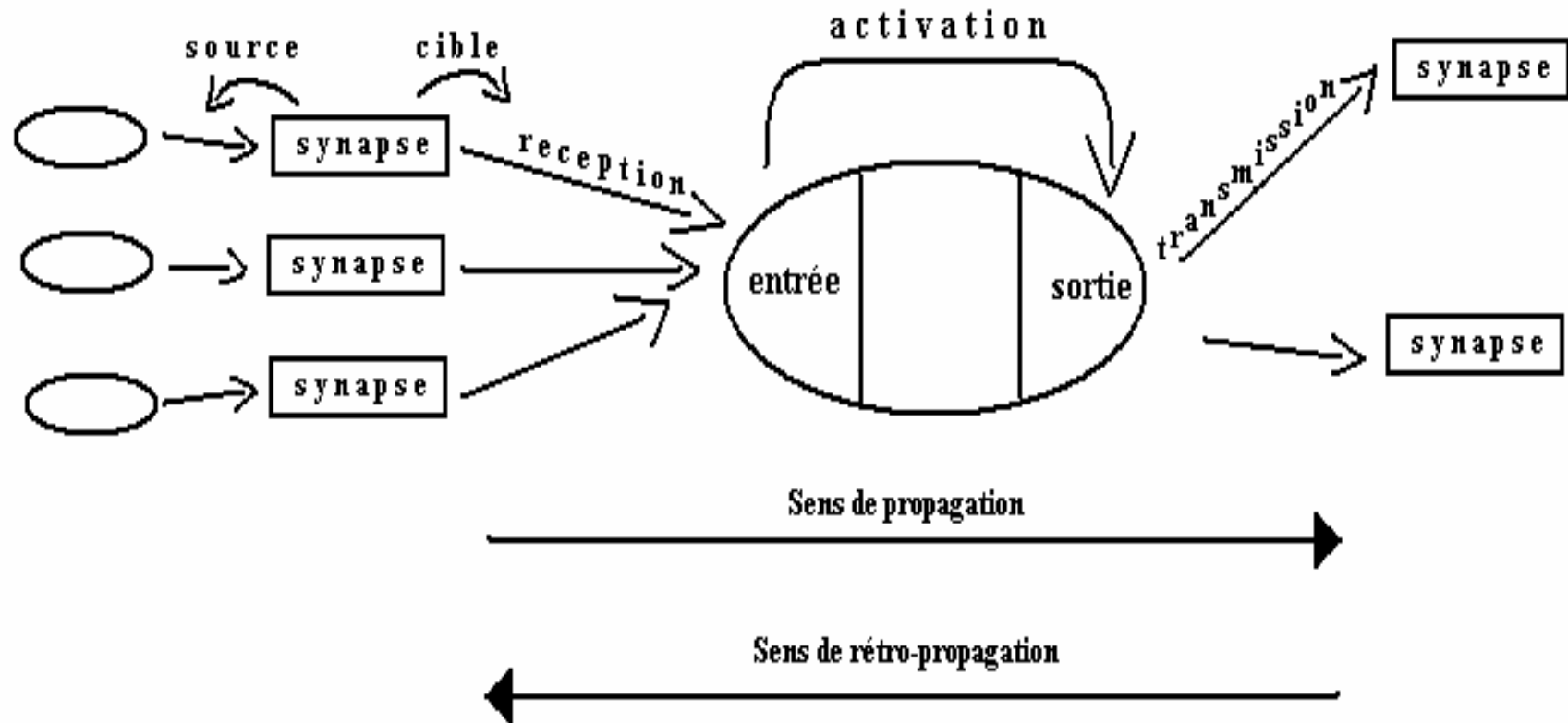
- extraire les micro-lignes dans l'image dans 4 directions
- connectivité "verticale" depuis le gradient
- connectivité "horizontale" = relaxation



- ⊓ Neurone sensible à une ligne verticale
- ⊓ Neurone sensible à une ligne horizontale
- connexion excitatrice
- connexion inhibitrice

# Le modèle de neurone utilisé

- Connecté à des synapses en amont et aval
- Neurones différenciés en fonction de l'aire



# L'apprentissage (1/3)

- Erreur quadratique= $\| \text{sortie} - \text{sortie désirée} \|^2$
- But de l'apprentissage : minimiser l'erreur quadratique sur l'ensemble des exemples
- Moyen : modifier poids du réseau
- 2 règles en parallèle :
  - rétro-propagation du gradient suivie de modification des poids
  - règle de Hebb

## L'apprentissage (2/3)

- Propagation en sens inverse d'un signal d'erreur, depuis la sortie jusqu'à la rétine : c'est la rétro-propagation du gradient de l'erreur
- Modification des poids proportionnellement au signal d'erreur
- signal d'erreur reste local !

# L'apprentissage (3/3)

- Règle de Hebb : issue d'observations biologiques ; cf : réponse conditionnée du chien de Pavlov
- Principe : renforcer connexions efficaces, inhiber connexions inefficaces

# Spécificité de notre approche

- Réseau inspiré du système de vision humain
- Organisation hiérarchique en aires spécialisées
- Interprétation abstraite de l'image (contours globaux)
- Neurones différenciés
- Apprentissage partiel (poids fixes, poids variables)
- Reconnaissance non spécifique à 1 type d'images

# Présentation des tests

## Introduction

1. Description des tests de fonctionnement
2. Taux de reconnaissance du réseau
3. Spécificité de la reconnaissance

**Conclusion:** améliorations à apporter

# Mesure du fonctionnement

- Habituellement, 3 taux décrivent un *classificateur*:
  - **Taux de réussite** : nombre d'images correctement classées, sur le nombre d'images montrées
  - **Taux de confusion** : nombre d'images rangées dans une mauvaise classe, sur le nombre d'images montrées en test
  - **Taux de rejet** : nombre d'images pour lesquelles le réseau ne se prononce pas, sur le nombre d'images montrées en test



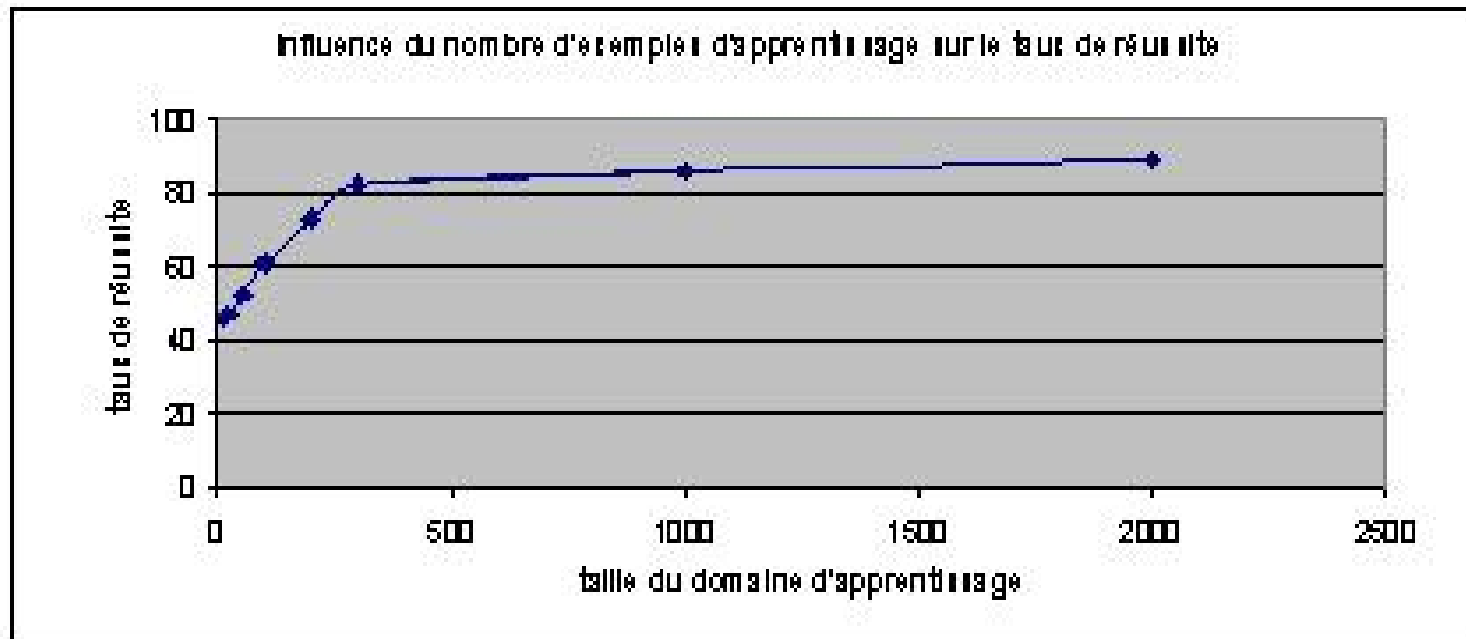
# La mesure utilisée

- Pour notre réseau : le rejet n'est pas permis
- Choix de notre part : rejet ou erreur nous sont indifférents
- Pour applications industrielles: un rejet demande réexamination par un expert
- mesure de fonctionnement choisie:  
**le taux de réussite**

# Rappel des objectifs

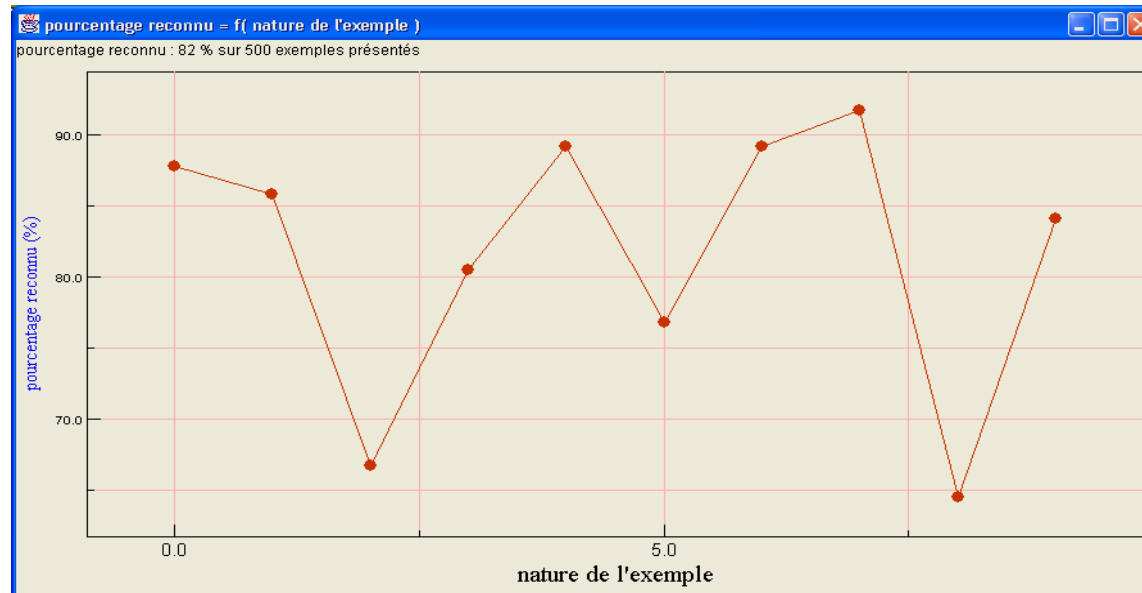
- Un taux de reconnaissance de 75 %
- Un faible nombre d'exemples d'apprentissage
- Images issues d'une base variée et largement utilisée dans le domaine de la reconnaissance manuscrite

# Taux de reconnaissance



- Un taux maximum qui plafonne à 90%
- Un taux de 75% dès 200 exemple appris

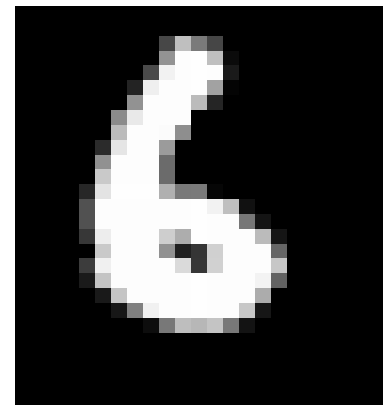
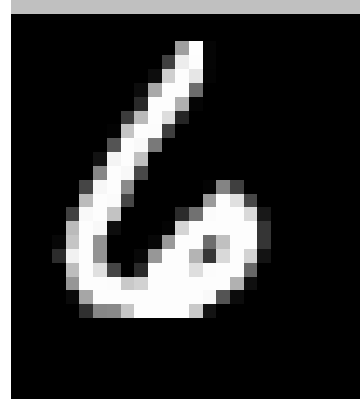
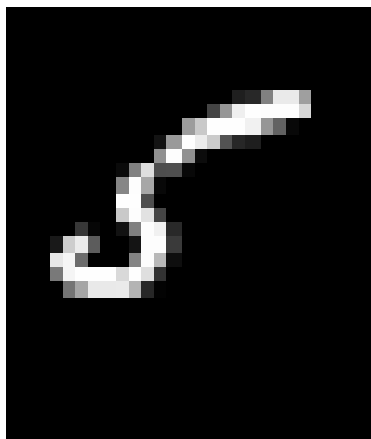
# Taux de réussite en fonction du chiffre présenté en entrée



- Variabilité de la reconnaissance de certains chiffres
- Fourchette de 65 % pour le « 8 » à 90% pour le « 7 »

# Explications de cette faiblesse

- Proximité de certains chiffres
- Ex: le « 5 » trop souvent pris pour un « 6 »



# Explications de cette faiblesse

- Meilleure reconnaissance des chiffres angulaires par rapport aux chiffres arrondis, plus variables
- Ex: le « 8 » et le « 2 »: 65% de taux de reconnaissance
- Ex: le « 1 » et le « 7 »: 90% de taux de réussite



# Conclusion de la partie scientifique

- Résultats encourageants
- Mais non définitifs: trop de poids à ajuster par rapport aux nombres d'exemples d'apprentissage
- Amélioration possible: utiliser la *régularisation* (imposer des contraintes aux poids )

# Prolongements possibles de ce projet

- Etudier en détail l'influence de certains paramètres  
Exemple : seuil de convergence
- Prendre en compte de manière plus satisfaisante les chiffres arrondis
- Intégrer les parties collatérales
  - neurones a codage temporel
  - apprentissage non supervisé de filtres



PRESENTATION DU

# **TRAVAIL EN GROUPE**

- 1 - Le déroulement du travail
- 2 - Difficultés rencontrées
- 3 - Leçons à en tirer

# **1 - Le déroulement du travail**

**Trois périodes :**

Septembre - Décembre

Janvier - Mars

Avril - Mai

## Septembre - Décembre

**But** : établir les bases et se familiariser avec le sujet

- Formation du groupe sur un thème vaste
- > Deux axes sont apparus :
  - reconnaissance de formes
  - réseaux de neurones
- Définition précise du sujet et des objectifs
- Rencontres avec des chercheurs :

Jean-Pierre NADAL (laboratoire de physique statistique de l'ENS ULM)

Philippe GAUSSIER (laboratoire de traitement du signal et de l'image de l'ENSEA)

Jean-Sylvain LIENARD et Philippe TARROUX (CNRS d'Orsay)

Nicolas BRUNEL (Université Paris V)

Yves FREGNAC et Jean LORENCEAU (CNRS de Gif-sur-Yvette)

Gérard DREYFUS, notre tuteur, du laboratoire d'électronique de l'ESPCI

## Janvier - Mars

**But** : travail personnel

- Définition des rôles
- > travail autour de petits groupes variables
  - le programme principal (~ 4/5 personnes)
  - annexes (~ 2/3 personnes)
  
- Spécialisation progressive

## Avril - Mai

**But** : atteindre les objectifs => efficacité

- Recentrage autour du programme principal
- > Deux parties en parallèle :
  - fin de la programmation
  - tests
- Spécialisation définitive sur les points de programmation

# 2 - Difficultés rencontrées

## Gérer la liberté

- liberté au niveau scientifique
- liberté au niveau du temps

## Se coordonner

- problèmes d'emplois du temps
- > Respecter les motivations

# 3 - Leçons à en tirer

- la motivation
- une définition appropriée des objectifs
- un planning rigoureux, des échéances précises
- une gestion personnalisée des membres du groupe
- le rôle d'un médiateur