

Introduction

L'étape de l'extraction des paramètres représente le cœur du système de reconnaissance, elle consiste à effectuer le traitement de l'image dans un autre espace de travail plus simple et qui assure une meilleure exploitation de données, et donc permettre l'utilisation, seulement, des informations utiles, dans ce chapitre nous allons étudier la méthode «Motifs binaires locaux (LBP) ».

Dans la deuxième section de ce chapitre nous nous sommes intéressés aux algorithmes de classification qui cherchent à discriminer entre les classes selon leur similarités (dissimilaires) de caractéristiques générées par la technique LBP pour chaque classe

Motifs binaires locaux (LBP pour Local Binary Pattern)

Le LBP de base a été développé lors de la visite de quelques mois de David Harwood de l'Université du Maryland à Oulu en 1992. Un point de départ pour la recherche était l'idée que les textures bidimensionnelles peuvent être décrites par deux mesures locales complémentaires : le motif et le contraste.

La LBP a été publiée pour la première fois dans le cadre d'une étude comparative des opérateurs de texture lors de la conférence internationale sur la reconnaissance des motifs (ICPR 1994), et une version étendue de celle-ci dans le journal de reconnaissance des modèles. La relation de LBP à la méthode de spectre de texture proposée par Wang et He a été trouvée lors de l'écriture du premier article sur LBP. Des années plus tard, il a également été constaté que LBP développé pour l'analyse de texture est très similaire à la transformation du recensement qui a été proposée à peu près au même moment que LBP pour l'informatique des correspondances visuelles en correspondance stéréo. [21]

1. Principe de la méthode

Les motifs binaires locaux (local binary patterns en anglais) sont des caractéristiques utilisées en vision par ordinateur pour reconnaître des textures ou pour la détection d'objet dans des images numériques.

Les motifs binaires locaux consistent à attribuer à chaque pixel P de l'image $I(i, j)$ à analyser, une valeur caractérisant le motif local autour de ce pixel. Ces valeurs sont calculées en comparant le niveau de gris du pixel central P aux valeurs des niveaux de gris des pixels voisins. [17]

L'opérateur LBP, introduit à l'origine par Ojala et al est une façon puissante de décrire la texture d'une image. La version originale de l'opérateur de modèle binaire local fonctionne dans un bloc de 3×3 pixels d'une image. Les pixels de ce bloc sont un seuil par sa valeur de pixel central, multiplié par des puissances de deux puis résumé pour obtenir une étiquette pour le pixel central. Comme le quartier se compose de 8 pixels, un total de $2^8 = 256$ étiquettes différentes peut être obtenu en fonction des valeurs de gris relatives du centre et des pixels dans le voisinage.

Les différentes étapes de la méthode LBP sont fournies au-dessous:

- L'image est d'abord divisée en cellules, constituée d'un nombre de blocs de 3×3 pixels. Le pixel central, px_c , dans un bloc est comparé par rapport à chacun de ses 8 voisins px_i ($i = 0, 1, \dots, 7$). Les pixels sont suivis le long d'un cercle, dans le sens des aiguilles d'une montre ou dans le sens inverse des aiguilles d'une montre....

$$LB(px_i - px_c) = \begin{cases} 1, & px_i \geq px_c \\ 0, & px_i < px_c \end{cases} \quad (3.1)$$

Lorsque tous les voisins prendront alors une valeur "1" si leur valeur est supérieure ou égale au pixel courant et "0" autrement. Cela donne un nombre binaire de 8 chiffres. Illustre l'opérateur LBP basique. [26]

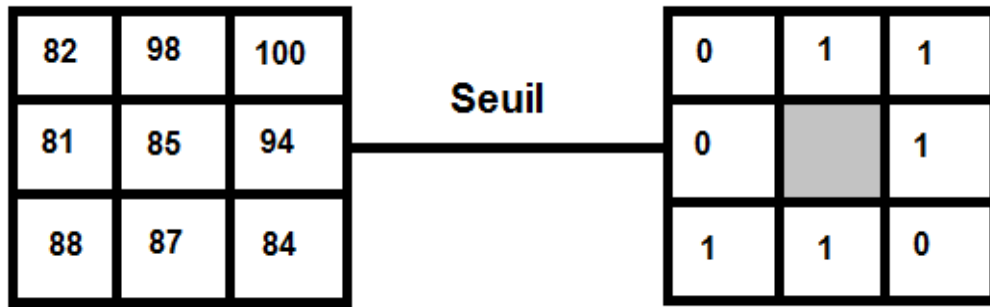


Figure 3.1 LBP basic.

- Ensuite, les valeurs des pixels obtenus après la comparaison sont multipliées par les poids donnés aux pixels correspondants. Enfin, les valeurs des huit pixels sont additionnées pour obtenir le numéro LBP pour le pixel central:

$$LBP = \sum_{i=0}^7 LB(px_i - px_c) \cdot 2^i \quad (3.2)$$

Le nombre LBP dérivé est 110 (2 + 4 + 8 + 32 + 64), illustré dans la Figure. 3.2 [26]



Figure 3.2 calcul du nombre LBP.

Cet opérateur peut également être étendu à des voisinages de dimensions différentes (Figure 3.3). Pour cela, un voisinage circulaire noté (P, R) est défini. Ici, P représente le nombre de points d'échantillonnage et R le rayon de la région.

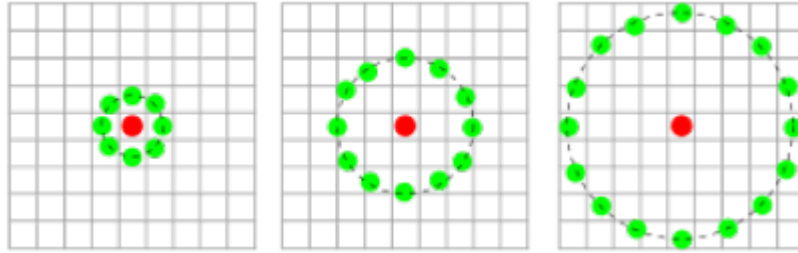


Figure 3.3 Trois voisinages circulaires: (8,1), (16,2), (24,3).

Ces points d'échantillonnage autour de pixel (x, y) se trouvent au point de coordonnées

$$(x_p, y_p) = (x + R \cos(2\pi/P), y - R \sin(2\pi/P)) \quad (3.3)$$

Quand un point d'échantillonnage ne tombe pas sur une coordonnée entière, la valeur de pixel est interpolée de manière bilinéaire. L'étiquette de LBP pour le pixel central (x, y) de l'image f (x, y) est obtenue par [21]:

$$LBP_{P,R}(x, y) = \sum_{p=0}^{P-1} s(f(x, y) - f(x_p, y_p)) 2^p \quad (3.4)$$

Avec

$$s(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0 \end{cases} \quad (3.5)$$

2. Mappages des étiquettes LBP: Modèles uniformes

Dans de nombreuses applications d'analyse de texture, il est souhaitable d'avoir des fonctionnalités invariantes ou robustes aux rotations de l'image d'entrée. Comme le LBPP, les motifs R sont obtenus par prélèvement circulaire autour du pixel central, la rotation de l'image d'entrée a deux effets: chaque voisinage local est tourné vers un autre emplacement de pixel, et dans chaque quartier, les points d'échantillonnage sur le cercle entourant le point central sont tournés dans une orientation différente. [21]

Une autre extension à l'opérateur d'origine utilise les mêmes motifs uniformes. Pour cela, on utilise une mesure d'uniformité d'un motif: U ("pattern") est le nombre des transitions entre les bits de 0 à 1 ou vice versa lorsque le motif de bits est considéré comme circulaire. Un motif binaire local est appelé uniforme si sa mesure d'uniformité est au maximum 2. Par exemple, les motifs 00000000 (0 transitions), 01110000 (2 transitions) et 11001111

(2 transitions) sont uniformes alors que les motifs 11001001 (4 transitions) et 01010011

(6 transitions) ne sont pas. En carte LBP uniforme, il existe une étiquette de sortie pour chaque motif uniforme et tous les motifs non uniformes sont attribués à une seule étiquette. Ainsi, Le nombre d'étiquettes de sortie différentes pour le mappage pour les motifs de P bits est

$P(P-1) + 3$. Par exemple, le mappage uniforme produit 59 étiquettes de sortie pour les quartiers de 8 points d'échantillonnage et 243 étiquettes pour les quartiers de 16 points d'échantillonnage. [21]

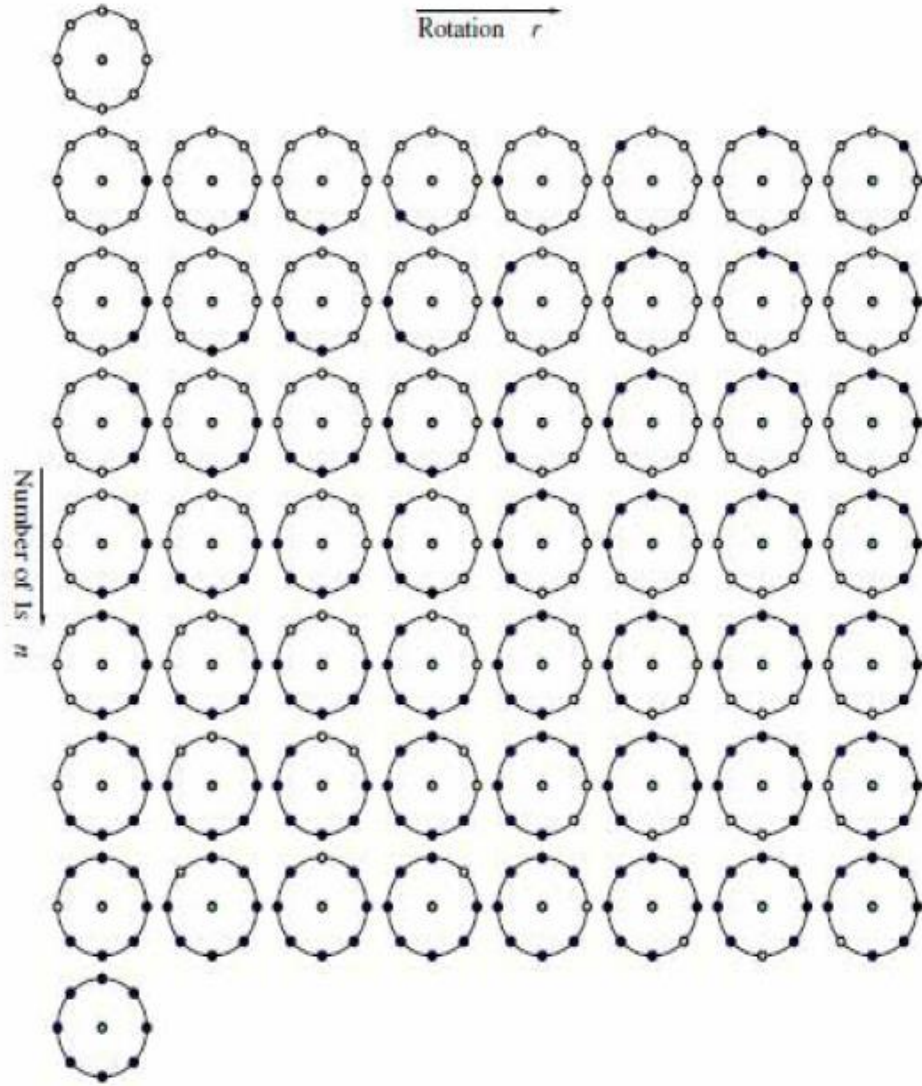


Figure 3.4 Les 58 motifs uniformes (8, R).

3. Invariance de rotation

On désigne un modèle spécifique de LBP uniforme par UP (n, r). Le couple (n, r) spécifie un modèle uniforme de sorte que n est le nombre de 1-bits dans le motif (correspondant au numéro de ligne dans la Figure 3.4) et r est la rotation du motif (numéro de colonne dans la Figure 3.4).

Maintenant, si le voisinage a pour P points, n prend une valeur de 0 à P+1, où $n=P+1$ est l'étiquette spéciale correspondant à tous les motifs non-uniformes. Et lorsque $1 \leq n \leq P-1$ la rotation de motif appartient à l'intervalle $0 \leq r \leq P-1$.

Soit $I^{\alpha^{\circ}}(x, y)$ est la rotation de l'image $I(x, y)$ par α degrés. Sous cette rotation, un point (x, y) est mis en rotation à (x', y') . Si on place un voisinage circulaire sur les points $I(x, y)$ et $I^{\alpha^{\circ}}(x', y')$ on observe qu'il tourne aussi avec un angle α° . [21]

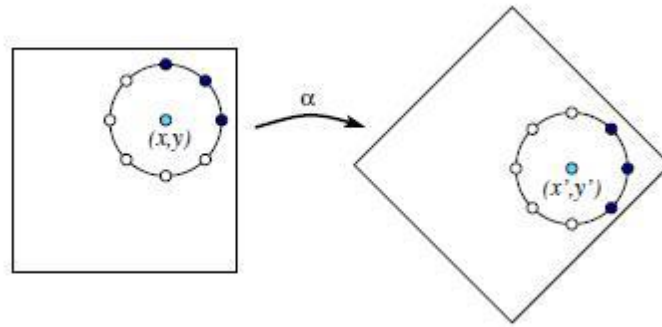


Figure 3.5 Effet de la rotation d'image sur le voisinage circulaire.

Si les rotations sont limitées à un entier multiplié par l'angle entre deux points, par exemple : $\alpha = a \cdot 360^\circ/P$, $a = 0, 1, \dots, P-1$. Le voisinage va tourner exactement selon un nombre de pas discret. Donc le motif uniforme $Up(n, r)$ au point (x, y) est remplacé par le motif uniforme $Up(n, r+a \bmod P)$ au point (x', y') de l'image après la rotation. [21]

Les méthodes de classification

1. Réseaux de neurones

Les réseaux de neurones artificiels (RNA) sont des réseaux fortement connectés de processeurs une sortie unique sur la base des informations qu'il reçoit. Ils infèrent des propriétés émergentes permettant de solutionner les problèmes jadis qualifiés de complexe.

Dans un réseau, chaque sous-groupe fait un traitement indépendant des autres et transmet le résultat de son analyse au sous-groupe suivant. L'information donnée au réseau va donc se propager de la couche d'entrée à la couche de sortie, en passant soit par aucune, une ou plusieurs couches intermédiaires. Il est à noter qu'en fonction de l'algorithme d'apprentissage.

1.1. Neurone biologique

Le neurone est une cellule composée d'un corps cellulaire et d'un noyau. Le corps cellulaire se ramifie pour former ce que l'on nomme les dendrites. Celles-ci sont parfois si nombreuses que l'on parle alors de chevelure dendritique ou d'arborisation dendritique. C'est par les dendrites que l'information est acheminée de l'extérieur vers le soma, corps du neurone.

L'information traitée par le neurone chemine ensuite le long de l'axone (unique) pour être transmise aux autres neurones. La transmission entre deux neurones n'est pas directe. En fait, il existe un espace intercellulaire de quelques dizaines d'Angstrom (10^{-9} m) entre l'axone du neurone afférent et les dendrites (on dit une dendrite) du neurone efférent. [29]

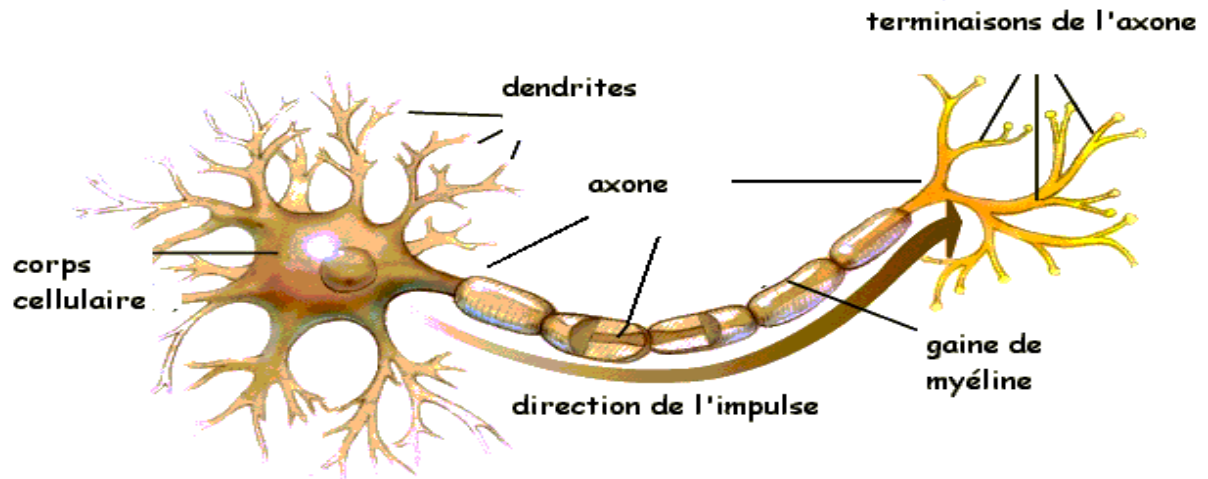


Figure 3.6 Neurone biologique.

Ainsi les axones des neurones (sorties) sont connectés aux dendrites d'autres neurones par les synapses, ce qui fait de notre cerveau un réseau extrêmement complexe de neurones, compte tenu que nous possédons plusieurs milliards de neurones et que chaque neurone peut avoir plusieurs milliers de dendrites. Les synapses jouent un rôle important car elles sont le lien entre la sortie d'un neurone et l'entrée d'un autre neurone.

L'influx nerveux passe par ces synapses, qui peuvent inhiber ou laisser passer le signal nerveux. Si l'amplitude du signal arrivant à la synapse est assez élevée, alors elle laissera passer le signal, sinon il sera inhibé. La synapse est capable de s'adapter pour laisser passer le signal ou empêcher son passage : ce mécanisme d'adaptation est à la base de l'apprentissage de notre cerveau. [29]

1.2. Neurone formel

Un "neurone formel" (ou simplement "neurone") est une fonction algébrique non linéaire et bornée, dont la valeur dépend de paramètres appelés coefficients ou poids. Les variables de cette fonction sont habituellement appelées "entrées" du neurone, et la valeur de la fonction est appelée sa "sortie".

Un neurone est donc avant tout un opérateur mathématique, dont on peut calculer la valeur numérique par quelques lignes de logiciel. On a pris l'habitude de représenter graphiquement un neurone comme indiqué sur la Figure 3.7. [16]

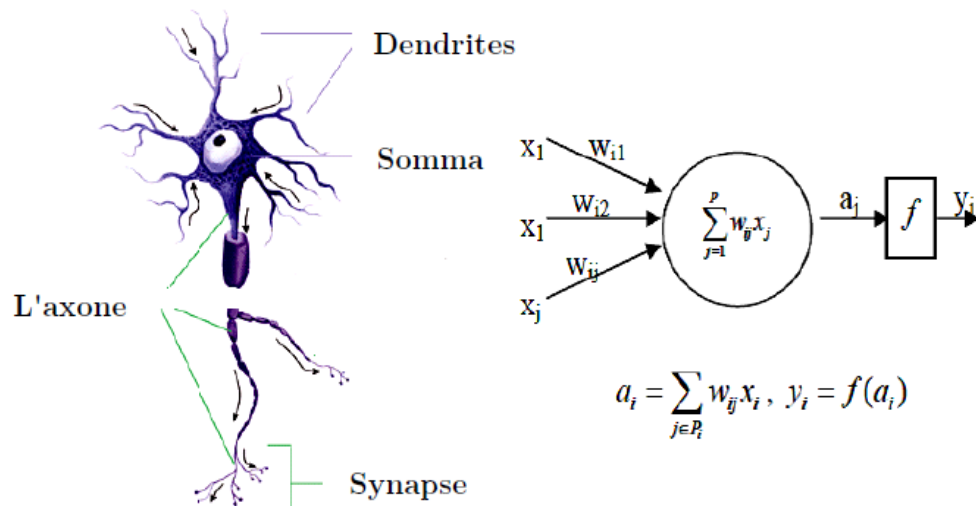


Figure 3.7 Neurone biologique et neurone formel.

Avec : x_i : paramètres d'entrées.

w_{ij} : poids synaptiques.

a_i : potentiel neuronal.

y_i : sortie du neurone i .

f : la fonction d'activation.

Il existe de nombreuses formes possibles pour la fonction d'activation. Les plus courantes sont présentées sur la (Figure 3.8). On remarquera qu'à la différence des neurones biologiques dont l'état est binaire, la plupart des fonctions d'activations sont continues, offrant une infinité de valeurs possibles comprises dans l'intervalle $[0, +1]$ ou $[-1, +1]$.

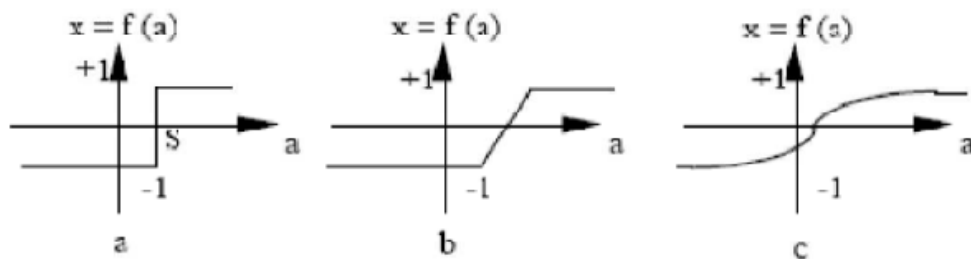


Figure 3.8 Différents types de fonctions d'activation pour le neurone formel,

a : fonction à seuil (S : valeur du seuil), b : linéaire par morceaux, c : sigmoïde.

Une collection de neurones formels est appelée réseaux de neurones. Ces réseaux sont souvent constitués de couches de neurones interconnectés.

1.3. Architecture des réseaux de neurones formels

On distingue deux grands types d'architectures de réseaux de neurones : les réseaux de neurones non bouclés et les réseaux de neurones bouclés.

1.3.1. Réseaux non bouclée (Statique)

Réseaux d'avance permettent aux signaux de voyager d'une manière unique; De l'entrée à la sortie. Il n'y a pas de commentaires (boucles), c'est-à-dire que la sortie de n'importe quelle couche n'affecte pas cette même couche. Les réseaux d'avance ont tendance à être des réseaux directs qui associent les entrées aux résultats. Ils sont largement utilisés dans la reconnaissance de motifs. Ce type d'organisation est également appelé Bas - En haut ou Haut - vers le bas. [39]

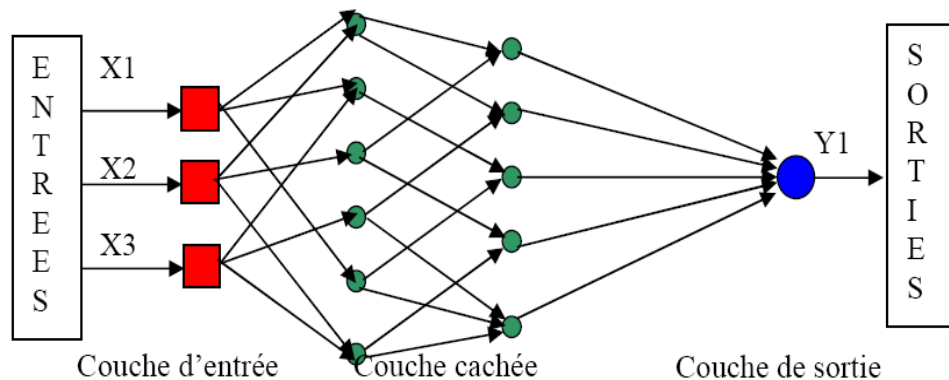


Figure 3.9 Réseau de neurone non bouclé.

1.3.2. Réseaux bouclée (Dynamique)

Les réseaux de rétroaction peuvent transmettre des signaux dans les deux sens en introduisant des boucles dans le réseau. Les réseaux de rétroaction sont très puissants et peuvent devenir extrêmement compliqués. Les réseaux de rétroaction sont dynamiques; Leur «état» change continuellement jusqu'à ce qu'ils atteignent un point d'équilibre. Ils restent au point d'équilibre jusqu'à ce que l'entrée change et qu'un nouvel équilibre soit nécessaire. Les architectures de rétroaction sont également appelées interactives ou récurrentes, bien que ce dernier terme soit souvent utilisé pour désigner des connexions de rétroaction dans des organisations à une seule couche. [39]

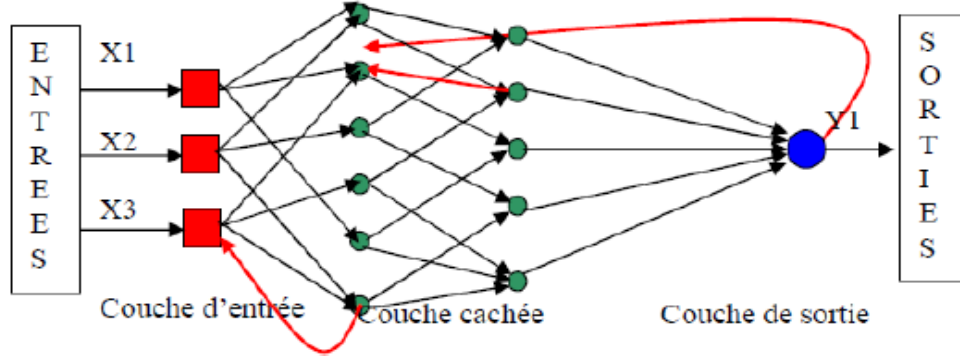


Figure 3.10 Réseau de neurone bouclé.

1.4. L'apprentissage des réseaux de neurones

On appelle « apprentissage » des réseaux de neurones la procédure qui consiste à estimer les paramètres des neurones du réseau, afin que celui-ci remplisse au mieux la tâche qui lui est affectée.

On peut distinguer deux types d'apprentissages : l'apprentissage « supervisé » et l'apprentissage « non supervisé »

1.4.1. L'apprentissage supervisé

L'apprentissage supervisé consiste à présenter au réseau une série d'exemples sur ses entrées, ainsi le réseau calcule sur ses sorties un vecteur d'état qui est comparé au vecteur des états désirés. Une erreur est calculée entre le vecteur des sorties désirées et le vecteur des sorties calculées par le réseau. L'algorithme d'apprentissage adapte les poids du réseau de façon à corriger son comportement comme le montre la figure 3.11. [6]

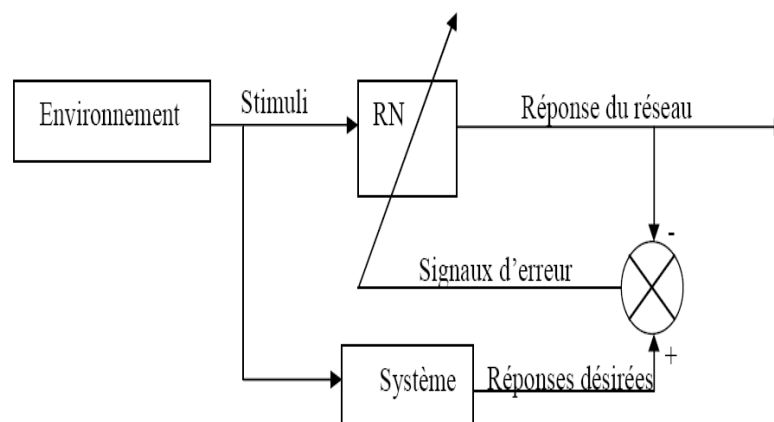


Figure 3.11 Apprentissage supervisé.

1.4.2. Apprentissage non supervisé

L'apprentissage est qualifié de non supervisé lorsque, seules les valeurs d'entrée sont disponibles. Il repose sur un critère interne de conformité du comportement du réseau par rapport à des spécifications générale et non sur des observations externes. Dans ce cas, l'apprentissage est basé sur des probabilités. Le réseau va se modifier en fonction des régularités statiques de l'entrée. L'algorithme d'apprentissage exploite pour se faire une mesure prédéterminée de la qualité de représentation de la connaissance afin d'ajuster les paramètres du réseau. Ce type d'apprentissage est représenté schématiquement en

Figure 3.12. [6]

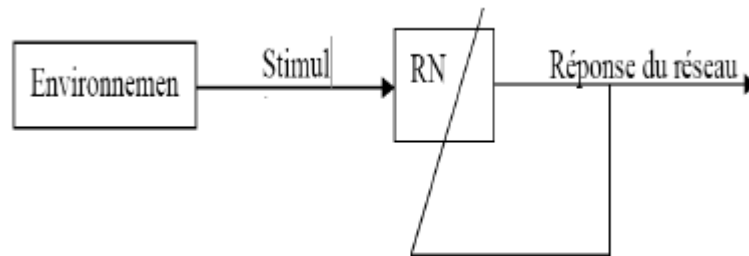


Figure 3.12 apprentissage non supervisé.

1.5. Le réseau de neurones probabiliste (PNN)

Le réseau de neurones probabiliste PNN (Probabilistic Neural Network) introduit par Donald Specht en 1988, est un réseau feed-forward à 3 couches utilisé pour la classification des données. À l'inverse des autres réseaux de neurones qui sont basés sur la rétro-propagation, le réseau de neurones probabiliste est basé sur des principes statistiques dérivés de la stratégie de décision de Bayes et des fonctions estimateurs de la densité de probabilité. Le PNN emploie des fonctions gaussiennes à base radiale et sphériques centrées à chaque vecteur d'apprentissage. [27]

Pour l'apprentissage d'un réseau probabiliste, il faut d'abord que chaque vecteur x en entrée soit normalisé de façon que : $\sum_{i=1}^n x_i^2 = 1$

Le nombre de neurones d'entrée est égal au nombre des paramètres (variables) de la forme à classifier. Pour chaque échantillon de la base d'apprentissage, on crée un neurone dans la couche cachée (couche de motifs) avec les connections correspondantes avec les neurones d'entrée de façon que $w_k = x_k$ pour $k=1, 2, \dots, n$. Ensuite une seule connexion est créée vers le neurone de la couche de classification correspondant à la classe de l'échantillon. Ce processus est illustré dans l'algorithme suivant : [1]

Début

Initialisation

$j=0$; n = premier échantillon d'apprentissage ;

Faire

$j \leftarrow j+1$;

Normalisation :

$$x_{jk} \leftarrow \frac{x_{jk}}{\sqrt{\sum_i^d x_{jk}^2}}$$

Apprentissage des poids : $w_{jk} \leftarrow x_{jk}$;

Si $x \in w_j$ **alors** $a_{ic} \leftarrow 1$;

Jusqu'à $j = n$;

1.6. Le réseau de neurones de fonctions à base radial (RBF)

La structure typique d'un réseau de neurones RBF (Figure 3.13) se compose de trois couches à savoir: une couche d'entrée, une couche cachée et une couche de rendement. Chaque couche se compose d'un certain nombre de neurones (nœuds). Dans la couche d'entrée, les nœuds sont employés pour faire passer seulement les données d'entrée à la couche cachée. Aucun calcul n'est exécuté dans les nœuds de la couche d'entrée. La couche cachée contient les k nœuds, qui appliquent une transformation non linéaire sur les variables d'entrées. [8]

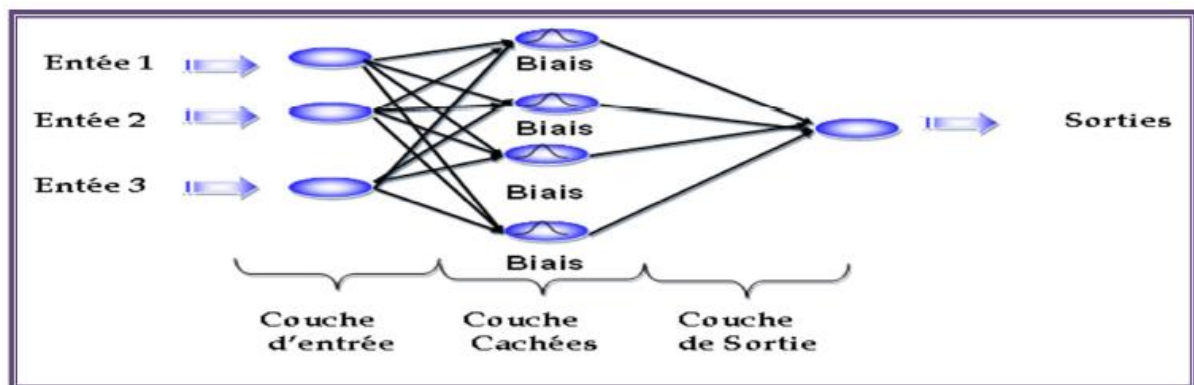


Figure 3.13 Architecture d'un réseau de neurone RBF.

1.7. Le réseau GRNN

Le réseau de neurones GRNN (*Generalized Regression Neural Network*) a une architecture assez semblable à celle des réseaux de neurones à base radiale (*Radial Basis Networks ou RBF*). Il possède deux couches de neurones. Les neurones de la couche cachent une fonction de transfert à base radiale et les deux neurones de la couche de sortie ont une fonction de transfert linéaire.

Tout comme le réseau MLP, le réseau GRNN doit être entraîné avant d'être utilisé. Pour l'entraîner, il faut fixer seulement le paramètre : « spread ». Plus le paramètre « spread » est faible, plus le neurone mémorise bien l'entrée qui lui a été assignée. Le réseau GRNN donne alors une excellente précision et une bonne capacité à mémoriser. [13]

2. Le classificateur du K plus proches voisins (KNN) :

C'est un classificateur simple basé sur le calcul de distance entre les exemples d'apprentissage et les exemples de tests, généralement la norme euclidienne est souvent employée comme mesure de distance, dans chaque étape de l'apprentissage, l'algorithme mémorise les k meilleurs exemples de l'ensemble d'apprentissage ($kppv(x)$) qui sont proches à l'exemple de test x. Cet algorithme est souvent performant s'il y a suffisamment d'exemples d'apprentissage, mais demande un temps de prédiction très long pour passer tous les exemples afin de trouver les K meilleures solutions. [27].

L'algorithme

Soit $L = \{(x', c) | x' \in R_d, c \in C\}$ est un ensemble d'apprentissage.

(x' est un exemple d'apprentissage et c sa classe qui lui est associée.

Soit x l'exemple de test dont on souhaite déterminer sa classe

Début

Pour chaque (*exemple* (x', c) $\in L$) **faire**

Calculer la distance $D(x, x')$

Fin

Pour chaque $\{x' \in kppv(x)\}$ **faire**

Compter le nombre d'occurrences de chaque classe

Fin

Attribuer à x la classe la plus fréquente

Fin

Conclusion

Quelque soit un system reconnaissance de formes, l'étape d'extraction des caractéristiques est une étape cruciale. En effet, elle permet de retenir que les informations pertinentes pour la phase de classification, par exemple dans ce manuscrit nous avons utilise, le motif binaire locale (LBP).

La classification est une tache qui consiste trouver un espace lequell permet discriminer les données selon leurs caractéristiques. Dans ce mémoire nous avons focalises sur Les méthodes de classification neuronales à savoir: les réseaux de neurones (PNN, RBF, GRNN).