

N° d'ordre : 1799

THESE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITE

spécialité : *Electronique*

par

Ludovic LELAURIN



Contribution à la définition d'un modèle générique de réseaux de neurones pour la minimisation de fonctionnelles. Application à l'estimation du mouvement dans des séquences d'images.

Soutenue le 27 septembre 1996 devant la commission d'examen composée de :

MM.

<i>J.P. Dubus</i>	<i>Président,</i>	<i>Co-directeur de Thèse</i>
<i>I. Magnin</i>		<i>Rapporteur</i>
<i>M. Roussel</i>		<i>Rapporteur</i>
<i>V. Devlaminck</i>		<i>Co-directeur de Thèse</i>
<i>M. Benjelloun</i>		<i>Examineur</i>
<i>F. Dufrenois</i>		<i>Examineur</i>

Remerciements

Je tiens à exprimer ici mes remerciements aux membres du jury :

Monsieur le Professeur J.P. DUBUS, Directeur du Laboratoire de Mesures Automatiques, qui m'a fait l'honneur de présider le jury de cette thèse et qui a bien voulu m'accueillir dans son laboratoire.

Madame I. MAGNIN, Directeur de recherche I.N.S.E.R.M. au CREATIS à l'I.N.S.A. de LYON et Monsieur M. ROUSSEL, Professeur à l'I.U.T. de Troyes, Directeur du Laboratoire de Traitements des Images, qui ont accepté de rapporter ce travail.

Messieurs M. BENJELLOUN, Professeur à l'Université du Littoral et F. DUFRENOIS, Maître de Conférences à l'Université du Littoral, pour leur participation à ce jury et pour l'intérêt qu'ils portent à ce travail.

Monsieur V. DEVLAMINCK, Maître de Conférences, qui est à l'origine de ce travail, pour les longs moments de disponibilité qu'il m'a accordés afin de discuter des développements de mon travail ainsi que pour la relecture de ma thèse .

Je n'oublierai pas mon "pote" A. YACOUBI. Son optimisme à toute épreuve m'a permis d'outrepasser les difficultés rencontrées au cours de cette thèse. Un petit mot aussi pour ceux qui ont accepté de faire un léger détour pour avoir l'honneur de déjeuner avec votre serviteur.

J'ai eu le plaisir d'effectuer des vacances à l'I.U.T. de Mesures Physiques. Pour m'avoir permis d'achever cette thèse dans les meilleures conditions, je tiens à remercier particulièrement A.M. DUTHILLEUL.

Enfin, je ne voudrais surtout pas oublier l'ensemble du personnel du laboratoire que j'ai eu la chance de côtoyer durant ces trois années.

*"L'intelligence est la faculté à l'aide de
laquelle nous comprenons finalement
que tout est incompréhensible"*

La vie des termites, Maurice MAETERLINK

Table des matières

CHAPITRE 1. INTRODUCTION.....	8
CHAPITRE 2. VISION PAR ORDINATEUR, REGULARISATION ET OPTIMISATION.....	13
2.1. La vision par ordinateur.....	14
2.1.1. Introduction.....	14
2.1.2. La vision primaire.....	15
2.2 Rappel mathématique sur les problèmes mal-posés.....	16
2.2.1. Définition générale.....	16
2.2.2. Problèmes correctement posés.....	17
2.2.3. Problèmes inverses.....	17
2.3. Méthodes de régularisation.....	18
2.3.1. Régularisation variationnelle aux problèmes de vision.....	20
2.3.2. introduction.....	20
2.3.3. Exemple du mouvement.....	22
2.4. Optimisation.....	25
2.4.1. Les problèmes d'optimisation sans contraintes.....	26
2.4.2. Les problèmes d'optimisation sous contraintes.....	26
2.4.3. Le recuit simulé.....	28
2.5. Conclusion.....	31

CHAPITRE 3. LA NEUROMIMETIQUE 32

3.1. Historique.....33

3.2. Le neurone formel.....33

3.2.1. Neurone formel binaire.....38

3.2.2. Neurone formel à valeurs réelles.....40

3.2.3. Neurone formel probabiliste.....41

3.3. La topologie des réseaux.....42

3.3.1. Réseau à couches.....42

3.3.2. Réseau récursif.....43

3.3.3. Réseau bicouche rétroactif.....44

3.3.4. Réseau multicouche à compétition/coopération.....45

3.3.5. Réseau monocouche topologiquement ordonné.....45

3.4. Les réseaux hybrides.....46

3.5. La dynamique.....48

3.5.1. Temps discret.....48

3.5.2. Temps continu.....50

3.6. Conclusion.....50

CHAPITRE 4. LES RESEAUX RECURSIFS 54

4.1. Mécanique statistique.....55

4.1.1. Energie.....56

4.1.2. Température.....56

4.2. Verres de spins.....57

4.3. Les réseaux récursifs.....60

4.3.1. Le modèle discret.....60

4.3.1.1. Définitions.....60

4.3.1.2. Dynamique.....61

4.3.1.3. Energie.....	62
4.3.2. Modèle continu.....	63
4.3.2.1. Dynamique.....	63
4.3.2.2. Energie.....	64
4.3.3. Mémoires associatives.....	65
4.3.4. Réseau de Hopfield et Optimisation.....	68
4.3.4.1. La méthodologie.....	69
4.3.4.2Réseaux récurrents et recuit simulé.....	71
4.4. Conclusion.....	72
CHAPITRE 5. METHODOLOGIE NEURONALE POUR L'OPTIMISATION	73
5.1. Définition d'une image.....	74
5.2. Méthodologie neuronale proposé pour l'optimisation.....	74
5.2.1. Modèle de fonctionnelle à minimiser.....	74
5.2.2. Estimation des dérivées partielles.....	75
5.2.3. Représentation de la variable.....	79
5.2.4. Estimation des paramètres du modèle.....	82
5.2.5. Dynamique du réseau.....	84
5.3. Exemple de la restauration.....	86
5.4. Conclusion.....	88
CHAPITRE 6. ESTIMATION DU MOUVEMENT	89
6.1. Mesure du mouvement apparent.....	92
6.1.1. les méthodes différentielles.....	93
6.1.2. Le flot optique par Horn & Schunck.....	97
6.1.2.1. Estimation des dérivées partielles de la fonction intensité.....	97
6.1.2.2. Estimation du Laplacien du flot des vitesses.....	98
6.1.2.3. Solution itérative de la minimisation.....	99
6.2. Méthodologie neuronale proposée pour le flot optique.....	100
6.2.1. Codage du vecteur vitesse.....	100

6.2.2. Architecture et dynamique du réseau pour le flot optique.....	102
6.2.2.1. Architecture globale du réseau.....	102
6.2.2.2. La dynamique.....	103
6.2.2.3. La fonction d'énergie.....	105
6.2.3. La phase d'identification.....	107
6.3. Résultats expérimentaux.....	110
6.3.1. Séquence d'images de synthèse.....	110
6.3.1.1. Cas de la translation pure.....	111
6.3.1.2. cas de la rotation.....	117
6.3.1.3. cas d'un mouvement divergent.....	119
6.3.1.4. Paramètre de quantification et minimum d'énergie.....	119
6.3.1.5. Cas d'images bruitées.....	120
6.3.2. Séquences d'images réelles.....	122
6.4. Extension à des mouvements déformables.....	133
6.4.1. Nouveaux paramètres du réseau.....	134
6.4.2. Résultats expérimentaux.....	136
6.5. Conclusion.....	138
CHAPITRE 7. CONCLUSION GENERALE	140
ANNEXES	143
ANNEXES I. L'apprentissage.....	144
ANNEXES II. Phase d'identification.....	149
ANNEXES III. Variation d'énergie.....	153
ANNEXES IV. Approche de Horn & Schunk.....	158
BIBLIOGRAPHIE	162

Chapitre 1

Introduction

L'objectif de cette thèse est de présenter un modèle générique de réseaux de neurones pour la résolution de problèmes d'optimisation. Cette architecture est basée sur le modèle discret de Hopfield. Afin d'effectuer une analyse la plus complète possible de cette approche, l'application à l'estimation du mouvement apparent dans des séquences d'images monoculaires a été choisie.

Le développement scientifique et le progrès technique favorisent la résolution de problèmes ou tâches de plus en plus complexes, en faisant appel au potentiel intellectuel des chercheurs, ingénieurs, mathématiciens, et spécialistes dans divers domaines. La révolution informatique a, d'autre part, augmenté l'efficacité des moyens de calcul et a permis de devenir plus exigeant sur les modèles physiques développés.

La vision par ordinateur permet à partir de l'image d'un objet ou d'une scène réelle, d'en déduire des données adaptables par une machine. Son développement s'est accentué à partir des années soixante dix dès que l'on a pu disposer de capteurs fiables et de matériel

informatique performant. Il existe différents types de capteurs : linéaires, matriciels, caméra, laser, Ils peuvent ainsi saisir une image à deux dimensions, comme les caméras ordinaires, ou capter des informations relatives à l'objet dans l'espace tridimensionnel. Le traitement de ces images est une opération classique à l'issue de laquelle on dispose d'une description abstraite (géométrique, topologique, ensembliste, ...) de l'objet ou de la scène réelle. L'étape suivante consiste à interpréter ces données, et on a souvent recours à des techniques de l'intelligence artificielle. Dans le cadre de notre travail, nous nous arrêterons à l'étape du traitement de l'image.

Nombre de ces problèmes de traitement des images peuvent s'exprimer sous la forme d'une fonctionnelle à minimiser. Cependant, pour la majorité d'entre elles, leurs formulations aboutissent à la résolution d'un problème mal-posé au sens d'Hadamard. Il est donc nécessaire de recourir à des techniques de régularisation. La base de ces méthodes réside dans l'emploi d'une contrainte supplémentaire qui va permettre de construire une solution approchée au problème initial. La solution finale doit rendre minimale la nouvelle fonctionnelle caractérisant le problème régularisé. Cette technique correspond à la résolution d'un problème d'optimisation sans contraintes.

Beaucoup de méthodes ont été développées pour résoudre ce genre de problème. Elles sont issues des mathématiques, *méthode de gradient*, *relaxation*, ou de la physique statistique, *recuit simulé*, *approximation en champ moyen*, *recuit en champ moyen*. Ces dernières années, des recherches menées sur l'analyse et le fonctionnement du cerveau humain ont permis de déterminer des concepts et des principes fondamentaux qui peuvent être appliqués à différents problèmes et notamment aux problèmes d'optimisation. C'est ainsi que des approches neuronales performantes ont vu le jour.

Les réseaux de neurones artificiels se composent d'un grand nombre d'unités élémentaires opérant en parallèle; ces unités de calcul se désignent sous le terme de *neurones* ou *neurones formels*. La fonction réalisée par ces réseaux se détermine par la topologie des connexions qui existent entre les neurones, par le poids ou intensité de ces connexions et par le choix de la fonction de transfert du neurone (modèle de neurone). De tels systèmes connexionnistes peuvent assurer des tâches de haut niveau telles que la classification, la mémorisation associative, l'optimisation. Deux grandes familles de

réseaux sont en vérité à distinguer. D'une part ceux qui sont censés reproduire un modèle idéalisé permettant d'étudier le fonctionnement et les propriétés des systèmes nerveux biologiques; d'autre part, les réseaux, inspirés au départ par les connaissances biologiques, qui sont des systèmes de traitement dont l'objectif premier est de réaliser des tâches technologiques bien définies. La modélisation de cette dernière famille est alors plus simplifiée que la première. Cette thèse s'intéresse à cette seconde catégorie de réseaux.

Historiquement, les premiers travaux sur la neuromimétique remontent à la première moitié de ce siècle et à l'article "A logical calculus of the ideas immanent in nervous activity" de McCulloch et Pitts en 1943 [McCulloch & Pitts, 1943]. Dans leur modèle, chaque unité de traitement (ou neurone) calcule une somme pondérée des signaux fournis par l'ensemble des autres unités auxquelles elle est connectée et le résultat final est comparé à une valeur de seuil. Si le seuil est dépassé, le neurone est activé. Dans le cas contraire, le neurone est inactif. Dans les deux cas, cet état est propagé à travers le réseau pour les étapes ultérieures. Sous cet aspect très simple, les réseaux de neurones fonctionnent comme des systèmes itératifs discrets. A cette époque, les travaux de W.S. McCulloch et W. Pitts suscitèrent énormément d'intérêt puisqu'ils démontraient que toute fonction logique pouvait être accomplie par de tels réseaux. En résumé, leur article montre que des équations dynamiques très simples peuvent engendrer des comportements complexes à condition de les combiner et de les inclure dans un réseau de topologie appropriée.

Le parallélisme intrinsèque de ces réseaux offre une alternative très innovante aux machines séquentielles. Le traitement de l'information est en effet distribué à travers toute la structure. Il s'ensuit une plus grande robustesse vis-à-vis d'un dysfonctionnement d'un petit nombre d'éléments, ainsi qu'une relative insensibilité aux écarts entre les caractéristiques des différentes unités de calcul.

L'essor de la neuromimétique depuis une vingtaine d'années a apporté une grande diversité dans les topologies des réseaux qui se caractérisent par l'application visée. Une classification est proposée dans [Lipmann, 1987] [Marren, Harston, & Pap, 1991]. Dans cette thèse, nous ne considérerons qu'un seul type de réseau, référencé sous le nom de *réseau de Hopfield* [Hopfield, 1982]. Ce réseau possède une topologie récursive où à

chaque coup d'horloge le signal de sortie des neurones est rebouclé vers les entrées. J. Hopfield fut le premier à montrer que, sous certaines conditions, l'évolution dynamique des réseaux récurrents est gouvernée par une fonction d'énergie où les états stables du réseau correspondent à des minima locaux de cette fonction. Il en découle que ce modèle de réseaux est apte à résoudre des problèmes d'optimisation [Hopfield, 1985].

Comme de nombreux problèmes de traitement d'images sont modélisés en termes de fonction d'énergie pour la plupart quadratique, nous développons une *méthodologie neuronale* dont le principe de base consiste à représenter la solution du problème d'optimisation en termes d'états de neurone, qui sont les variables discrètes d'un espace euclidien. L'application à l'estimation du mouvement apparent permet de valider notre approche sur un problème complet puisque la solution recherchée est de type vectoriel, à valeur continue et possède une notion de dimension (relative à une vitesse).

Le chapitre II situe le problème. Nous rappelons qu'un grand nombre de problèmes de traitement des images peuvent s'exprimer sous la forme d'une fonction d'énergie. Ces problèmes sont à leur source mal-posés au sens d'Hadamard dont la définition mathématique est précisée. L'introduction d'une contrainte additionnelle de régularisation est alors requise. La nouvelle formulation du problème aboutit à la résolution d'un problème d'optimisation.

Le chapitre III présente une synthèse générale sur la neuromimétique. Nous rappelons la notion de neurone formel, présentons les différentes topologies de réseaux et dynamiques existantes.

Dans le chapitre IV, nous nous intéressons plus particulièrement aux réseaux récurrents. Nous montrons l'intérêt qu'a suscité ce modèle de réseau, en particulier chez les physiciens, de par son analogie avec les verres de Spin.

Le chapitre V, lui, expose de façon générale la méthodologie neuronale proposée pour la résolution de problèmes d'optimisation exprimés sous forme d'une fonctionnelle de type quadratique. Une première partie sera consacrée à l'approximation discrète des dérivées spatio-temporelles. Enfin, nous montrons que le

travail de Y.T. Zhou et R. Chellappa sur le problème de la restauration des images se trouve être un cas particulier de notre approche.

Le chapitre VI est dédié au problème de l'estimation du mouvement apparent dans des séquences d'images. Parmi les différentes approches d'estimation du mouvement, les méthodes différentielles s'inscrivent dans notre méthodologie. Nous rappelons succinctement l'état des diverses techniques classiques existantes, puis nous nous attachons ensuite aux méthodes différentielles existantes et en particulier à *l'équation de contrainte du mouvement apparent (E.C.M.A.)* pour aboutir à la notion de *flot optique*. Nous validons notre architecture sur différentes séquences d'images de synthèse et réelles en les comparant à l'approche classique de Horn & Schunk, préalablement décrite. Nous appliquons notre réseau sur un autre exemple correspondant au cas des mouvements d'objets déformables. La prise en compte de cette autre fonctionnelle se fait par une simple modification des valeurs de ses paramètres, poids synaptiques et entrées externes, tout en conservant son architecture précédente dû au caractère générique de notre démarche.

Chapitre 2

Vision par ordinateur, Régularisation et Optimisation

La vision par ordinateur s'intéresse à l'étude du traitement de l'information visuelle. Ses deux objectifs sont le développement d'outils d'interprétation des images et l'étude du système visuel humain.

La vision primaire se constitue d'une série de tâches visuelles qui a pour objectif d'extraire les propriétés physiques des surfaces visibles. Plusieurs problèmes de traitements d'images ont ainsi été résolus et différents algorithmes ont été développés avec succès. Des exemples concrets sont la stéréovision, l'estimation du flux optique, le calcul de la forme à partir des ombres, la reconstruction de surface, La majorité de ces tâches représentent les solutions de *problèmes inverses*. Cette caractéristique commune signifie que les problèmes de vision primaire sont des *problèmes mal posés* au sens d'Hadamard. Les mathématiques ont alors apporté leurs théories et leurs méthodes pour solutionner ce type de problèmes. Parmi ceux-ci, la construction de solutions *régularisées* permet d'obtenir une

solution approchée à la solution du problème initial. De nombreux problèmes de vision ont ainsi été abordés par des *méthodes de régularisation standard* qui imposent des contraintes sur la solution par un *principe variationnel* [Poggio, Torre, & Koch, 1985].

La transcription physique de ces problèmes de vision régularisés conduit en général à une fonction de forme quadratique, appelée *énergie du problème* ou encore *fonctionnelle*. Son minimum absolu correspond à la solution optimale du problème. La résolution de tels problèmes fait appel à des techniques d'*optimisation*. Les algorithmes qui en découlent, sont issus de travaux mathématiques voire de la mécanique statistique.

2.1 La vision par ordinateur

2.1.1 Introduction

L'étude de la vision a de tout temps suscité un grand intérêt. Ceci s'explique aisément si l'on sait que, chez l'homme, 60% du cortex cérébral est dévolu à ce sens, et que l'oeil est l'organe le plus complexe et le plus sophistiqué chez les vertébrés supérieurs. Aujourd'hui, nous savons que la vision comprend deux phases distinctes : la perception (au niveau de l'oeil) et le traitement (au niveau du cerveau) [Rémy, 1994]. Cette connaissance de la vision humaine a servi de modèle pour la réalisation de systèmes de vision artificiels. La mise en oeuvre de ces systèmes artificiels capables d'accomplir automatiquement ces fonctions a donné naissance à une nouvelle technique et un néologisme : *la visionique*. Ce terme désigne ainsi l'ensemble des techniques et méthodes qui participent à la formation, à l'acquisition et au traitement des images en vue de prendre une décision de façon automatique. Elles sont généralement implantées sur ordinateur, aussi parle-t-on de *vision par ordinateur*.

Un système de vision par ordinateur comprend :

- un ou plusieurs capteurs de lumière ou caméras qui "photographient" des objets ou des scènes, à l'instar de l'oeil humain.
- des processeurs et des programmes de traitement, correspondant à la zone du cerveau dédiée à la vision.
- entre les deux, un système de sélection, de numérisation et de transmission des données, qui jouerait le rôle du nerf optique.

Combinant optique, électronique, traitement du signal et informatique, la vision par ordinateur substitue au système visuel humain un ensemble informatisé (matériel+logiciel) capable d'intégrer l'acquisition d'images, leur traitement et la prise de décision adaptée au cas identifié. Cette analogie faite avec le système de vision humaine demeure néanmoins approximative, car ces systèmes artificiels sont loin d'atteindre la complexité de son original.

Toutefois, l'objectif de la vision artificielle n'est pas le même que celui de la vision biologique. L'être humain apprécie d'un coup d'oeil une image, mais il est incapable de l'exploiter de façon systématique. C'est la finalité de l'analyse et du traitement d'images informatiques. Le système se compose d'une partie capteur (l'oeil pour l'être humain), et d'une partie traitement et système d'interprétation de l'image (le cerveau chez l'être humain). Ces tâches sont remplies par différents processus regroupés sous le terme de *vision primaire*.

2.1.2 La vision primaire

La vision primaire peut se définir comme l'inverse de l'optique. Dans l'optique classique ou dans le graphisme par ordinateur, le problème fondamental est de déterminer les images des objets physiques, tandis que la vision est confrontée au problème inverse de retrouver des formes à trois dimensions à partir des images à deux dimensions.

Elle est constituée d'une série de tâches dont l'objectif est l'extraction des propriétés physiques des surfaces visibles à trois dimensions à partir des informations de luminance dans l'image à deux dimensions. Leurs résultats combinés correspondent grossièrement à la scène 2-1/2D de D. Marr [Marr, 1982] et aux images intrinsèques de H.G. Barrow et J.M. Tennenbaum [Barrow & Tennenbaum, 1981]. La liste ci-dessous fournit des exemples de tâches de vision primaire :

- Détection de contours
- approximation et interpolation spatio-temporelles
- calcul du flot optique
- la forme à partir des contours
- la forme à partir de la texture

- la forme à partir d'une ombre
- stéréovision
- la structure à partir du mouvement
- la structure à partir d'images stéréoscopiques
- reconstruction de surface
- calcul des couleurs d'une surface,

La majorité de ces tâches de vision primaire sont considérées comme les solutions de *problèmes inverses*. Cette caractéristique commune répond au formalisme mathématique *des problèmes mal-posés* au sens d'Hadamard [Poggio & Koch, 1985]. Leurs solutions dans la plupart des cas ne sont pas uniques, ou ne dépendent pas de façon continue des données dans le cas de la détection de contours [Torre & Poggio, 1986] [Poggio, Voorhees, & Yuille, 1985]. L'idée essentielle pour résoudre un problème mal-posé, c'est à dire pour le rendre "correctement-posé," est de restreindre la classe des solutions admissibles en introduisant une connaissance *a priori* appropriée. Cette connaissance *a priori* peut être exploitée, par exemple, sous la forme de propriétés statistiques de l'espace des solutions ou par des contraintes variationnelles qui imposent le domaine des solutions.

2.2 Rappel mathématique sur les problèmes mal-posés

Les problèmes de vision, mal-posés dans leur essence, font partie de la classe des problèmes mathématiques incorrectement posés [Badeva & Morozov, 1991].

2.2.1 Définition générale

Un grand nombre de problèmes physiques se ramène à la résolution d'une tâche mathématique du type :

$$A \cdot z = b, b \in Y \quad (2.1)$$

où $A : D_A \subseteq Z \rightarrow Y$, est un opérateur du domaine de définition D_A non vide, opérant de l'espace métrique Z à valeurs dans l'espace métrique Y . A peut être un opérateur intégral, différentiel ou algébrique linéaire ou non linéaire.

2.2.2 Problèmes directs

La définition des tâches correctement posées a été formulé pour la première fois par le mathématicien français J. Hadamard [Hadamard, 1902] :

$$1) \text{ Solubilité : } \forall b \in Y \Rightarrow \exists z \in D_A, A \cdot z = b$$

$$2) \text{ Non ambiguïté : } \forall z_1, z_2 \in D_A, A \cdot z_1 = A \cdot z_2 \Rightarrow z_1 = z_2$$

$$3) \text{ Stabilité : } \forall \bar{z}, \tilde{z} \in D_A, A \cdot \bar{z} = \bar{b}, A \cdot \tilde{z} = \tilde{b} \Rightarrow \bar{z} \rightarrow \tilde{z} \text{ dans } Z \text{ avec } \bar{b} \rightarrow \tilde{b} \text{ dans } Y$$

Les conditions 1) et 2) caractérisent le déterminisme mathématique du problème, alors que son déterminisme physique et la possibilité de sa résolution numérique sur la base des données sont associées à la troisième condition. En résumé, un problème est dit correctement posé si sa solution existe, est unique et dépend de manière continue de ses données.

Des exemples types de tâches incorrectement posées au sens d'Hadamard sont la tâche de Dirichlet et ses analogues, la tâche de Cauchy de l'équation de Laplace [Hadamard, 1932],

La plupart des problèmes de physique classiques sont correctement posés et J. Hadamard a démontré que ces problèmes devaient être correctement posés. Dans ces approches, il s'agit de déterminer la solution b à partir des données z . On parle alors de *problème direct*.

2.2.3 Problèmes inverses

Les rôles données-solutions sont dans ce cas inversés par rapport au problème direct. On cherche à déterminer des caractéristiques quantitatives d'objet z à partir de certains paramètres mesurés b . Z devient alors *l'espace des solutions du problème*. Ces problèmes font partie de la classe des *problèmes inverses*. Ils ne s'accommodent pas du concept "correctement posé" d'Hadamard et sont donc des problèmes mal-posés. Tels sont les problèmes de détermination des paramètres d'une équation différentielle, de tomographie par ordinateur, de restauration d'images et de définition de la forme d'un tambour à partir de sa fréquence de vibration, célèbre problème de M. Kac [Kac, 1966]. De même, les tâches de vision, comme il a été mentionné précédemment, sont des problèmes

mal posés puisque l'opérateur correspondant à A n'est pas, du point de vue mathématique, *injectif*.

La résolution de ces tâches incorrectement posées, est donc liée à la possibilité de construire des solutions adaptées. On peut citer en particulier la méthode de Lavrentiev [Lavrentiev, 1955], la méthode de quasi-solution [Ivanov,1962], les méthodes de régularisation, de déviation [Phillips, 1962], et de quasi-inversion, ou encore des méthodes statistiques.

2.2.4 Méthodes de régularisation

Les deux premières méthodes de M.M. Lavrentiev et V.K. Ivanov se caractérisent par la nécessité que l'ensemble des solutions de l'équation (2.1) soit compact. Mais en pratique, dans plusieurs cas de tâches, il ne l'est pas et de légères variations dans les données b peuvent entraîner de grandes variations dans la solution z. Leur résolution passe alors par la construction de solutions approchées, stables par rapport à ces petites variations. La base de cette méthode réside dans l'emploi d'un opérateur régularisant, d'où son nom de *méthode de régularisation* [Tikhonov, 1963].

On distingue ainsi trois grandes méthodes de régularisation standard :

a) parmi z qui satisfait $\|Pz\| \leq C$, où C est une constante, trouver z qui minimise

$$\|Az-b\| \quad (2.2)$$

b) parmi z qui satisfait $\|Az-b\| \leq \varepsilon$, trouver z qui minimise

$$\|Pz\| \quad (2.3)$$

c) trouver z qui minimise

$$\|Az-b\|^{2+\alpha} \cdot \|Pz\|^2 \quad (2.4)$$

où P exprime les contraintes physiques de la tâche et α est un paramètre de régularisation. A est un opérateur linéaire et P est linéaire.

La première méthode a) recherche la solution z qui satisfait la contrainte $\|Pz\| \leq C$ et approche au mieux les données. La seconde méthode b) estime la solution z qui est suffisamment proche des données b et est la plus "uniforme." Dans la méthode c), le

paramètre de régularisation α contrôle le compromis entre le degré de régularisation de la solution et sa proximité aux données. Des études théoriques de la régularisation ont permis de développer des techniques pour trouver la meilleure valeur du paramètre α possible [Tikhonov & Arsenin, 1977] [Wabba, 1980]. Elles fournissent aussi un vaste ensemble de résultats sur la forme de la fonctionnelle de stabilisation P adoptée pour assurer l'unicité de la solution et sa convergence. A.N. Tikhonov propose ainsi, pour s'assurer l'unicité de la solution à l'équation (2.1), des fonctionnelles de stabilisation (aussi appelée *stabiliseur d'ordre p*) que l'on peut exprimer sous la forme suivante :

$$\|Pz\| = \int \sum_{r=0}^p c_r(\xi) \cdot \left(\frac{d^r z}{d\xi^r} \right) \cdot d\xi \quad (2.5)$$

où $c_r(\xi)$ sont des facteurs de pondérations non négatifs. Ce modèle de stabiliseur procure des solutions semblables à celles que l'on obtiendrait par interpolation ou par approximation au moyen de splines.

Le choix des normes $\|\cdot\|$ et d'une fonctionnelle de stabilisation $\|Pz\|$ est dicté par des considérations mathématiques et, de manière plus importante, par une analyse physique des contraintes génériques sur le problème.

Par conséquent, les méthodes de régularisation *standard* permettent de construire des solutions approchées d'un problème mal-posé, en imposant des contraintes physiques sur ce dernier. Ce problème s'exprime alors sur le modèle de l'équation (2.4). L'étape suivante consiste donc à déterminer la solution de cette équation, communément appelée *énergie* du problème. Le résultat final doit rendre minimal la fonction énergie caractérisant le problème. Cette démarche correspond à la résolution d'un problème *d'optimisation*. Avant d'exposer les théories et techniques d'optimisation, nous exposons certaines tâches de vision primaire régularisées par des *contraintes variationnelles*.

2.3 Régularisation variationnelle aux problèmes de vision

2.3.1 introduction

La *régularisation variationnelle* [Poggio, Torre, & Koch, 1985] dénote les méthodes de régularisation qui reformulent un problème mal-posé, correctement-posé en terme de principes variationnels.

Les principes variationnels ont été très largement employés en physique, économie et sciences de l'ingénieur. En physique, par exemple, la plupart des lois fondamentales ont une formulation compacte en terme de principes variationnels qui requiert la minimisation de fonctionnelle adéquate, telle qu'une énergie ou un lagrangien dans le cas de la mécanique classique.

De même, dans les problèmes de vision primaire, des principes variationnels ont été utilisés pour introduire les contraintes physiques spécifiques. Par exemple, K. Ikeuchi et B.K.P. Horn [Ikeuchi & Horn, 1981] appliquent cette méthode pour reconnaître la forme d'un objet à partir des informations d'une ombre (*shape from shading*). Auparavant, K. Ikeuchi [Ikeuchi, 1980a] [Ikeuchi, 1980b] utilisa une technique similaire pour la reconnaissance de formes à partir de la texture (*shape from texture*). L'estimation du mouvement dans une image est obtenue en cherchant un champ de vitesse le plus uniforme, compatible avec les données [Horn & Schunk, 1981] [Hildreth, 1984a] : en d'autres termes, parmi tous les champs de vitesse possibles compatibles avec les données, le champ de vitesse où les variations sont les plus faibles sera choisi. De manière similaire, le problème de l'interpolation d'une surface, au travers des données stéréoscopiques éparses, peut être résolu en minimisant une fonction d'énergie qui intégrerait une contrainte de lissage [Grimson, 1981] [Terzopoulos, 1983]. L'évaluation de contours subjectifs [Ullman, 1976], de la luminance [Horn, 1974] et d'une forme à partir des contours [Brady & Yuille, 1984] peuvent eux aussi être formulé en terme de principes variationnels. D. Terzopoulos [Terzopoulos, 1984] a par la suite réexaminé ces techniques dans les problèmes de vision sous une approche plus théorique.

Tâche	Fonction d'énergie
Détection de contour	$\iint [(S \cdot i - f)^2 + \alpha \cdot (i_{xx})^2] \cdot dx$
approximation spatio-temporelle	$\iiint [(S \cdot i - f)^2 + \alpha \cdot (\nabla i \cdot \nabla + i_t)^2] \cdot dx \cdot dy \cdot dz$
Flux optique (basée sur les gradients)	$\iint [(i_x \cdot u + i_y \cdot v + i_t)^2 + \alpha \cdot (u_x^2 + u_y^2 + v_x^2 + v_y^2)^2] \cdot dx \cdot dy$
Flux optique (basée sur les contours)	$\iint [(V \cdot N - V^N)^2 + \alpha \cdot ((\partial/\partial s)V)^2] \cdot ds$
Reconstruction de surface	$\iint [(S \cdot f - d)^2 + \alpha \cdot (f_{xx}^2 + f_{xy}^2 + f_{yy}^2)^2] \cdot dx \cdot dy$
Couleur	$\ I^c - A \cdot z\ ^2 + \lambda \cdot \ P \cdot z\ ^2$
Stéréovision	$\iint \left\{ \left[\nabla^2 G * (L(x,y) - R(x + d(x,y), y)) \right]^2 + \alpha \cdot (\nabla d)^2 \right\} \cdot dx \cdot dy$
Forme à partir d'une ombre	$\iint [(E - CR(f, g))^2 + \alpha \cdot (f_x^2 + f_y^2 + g_x^2 + g_y^2)^2] \cdot dx \cdot dy$

Dans le premier exemple, la détection de contours [Torre & Poggio, 1986], les données sur l'intensité i ($i=i(x)$, cas monodimensionnel pour simplifier) sont distribuées sur un maillage discret: l'opérateur S est un opérateur d'échantillonnage sur la distribution continue f à estimer. Une fonction similaire peut être employée pour la tâche de l'approximation spatio-temporelle. L'intensité spatio-temporelle $f(x,y,t)$ se détermine à partir des données $i(x,y,t)$; le terme régularisant impose la contrainte d'une vitesse V constante dans le plan image. Dans le calcul du flux optique basé sur l'intensité, i représente l'intensité de l'image, u et v les deux composantes des vecteurs vitesse. Dans la reconstruction de surface, $f(x,y)$ est évaluée à partir des données de profondeurs $d(x,y)$. Dans le cas de la couleur, la luminance est calculée sur chacune des trois composantes I^c ($c=1, 2, 3$) qui correspondent en général aux trois filtres Red, Green, Blue; le vecteur solution z contient les composantes d'illumination et d'albedo de façon séparé; cette solution est appliquée par A sur les données idéales. Pour les tâches de stéréovision et de reconnaissance d'une forme à partir d'une ombre, les deux fonctionnelles régularisées sont non quadratiques. Pour la stéréovision, R et L caractérise respectivement les images de droite et gauche, et d , le champ de disparité; la régularisation du champ de disparité d comprend la convolution du laplacien d'une gaussienne G avec les images de droite R et de gauche L . Enfin, pour la reconnaissance d'une forme à partir d'une ombre, E représente la distribution de la luminance, CR , la carte des reflectances, et f et g sont relatifs aux composantes du gradient surfacique.

Tableau 1 : codage des tâches de vision primaire régularisées
(retranscrit d'après T. Poggio, V. Torre, & C. Koch, *Nature*, 317, pp 314-319, 1985)

Le Tableau 1, présente la formulation énergétique de certaines tâches de vision primaire. Nous allons à présent nous intéresser plus particulièrement au cas de l'estimation

du mouvement dans une séquence d'images. Nous nous inspirons pour cela des travaux de E.C. Hildreth [Hildreth, 1984a] [Hildreth, 1984b].

2.3.2 Exemple du mouvement

Une des tâches importantes de la vision primaire est l'estimation du mouvement. Le calcul d'un champ de vitesses à deux dimensions dans une image est une phase décisive dans plusieurs travaux tels que ceux destinés à recouvrer le mouvement et la structure tridimensionnelle des objets.

La détection de mouvement dans une séquence d'images nécessite l'existence d'une variation de l'intensité dans le plan image au cours du temps. La combinaison de ces deux paramètres (variation d'intensité-temps) est utilisé pour mesurer la direction et l'amplitude de la vitesse. Deux principaux schémas de mesures du mouvement sont à distinguer :

- ⇒ celui basé sur les dérivées spatio-temporelles de l'intensité lumineuse, la méthode *des gradients* [Fennema & Thompson, 1979].
- ⇒ celui basé sur les changements significatifs de l'intensité lumineuse, *les contours* [Marr & Ullman, 1981].

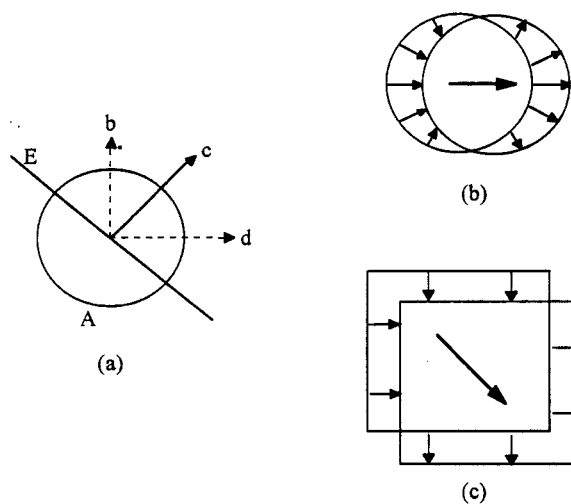


Figure 1 : ambiguïté du mouvement

(redessinée d'après E.C. Hildreth, *Proc. R. Soc. Lond. B*, 221, pp 189-220, 1984)

L'estimation du mouvement est un problème ambigu. Les mesures locales du mouvement, obtenues directement à partir des variations d'intensité dans l'image, ne

fournissent en général que la composante perpendiculaire du vecteur vitesse comme l'illustre la Figure 1. Si le mouvement de la droite E, Figure 1.a, devait être estimé par un détecteur à l'intérieur de la zone A, seule la composante c perpendiculaire à la droite serait extraite.

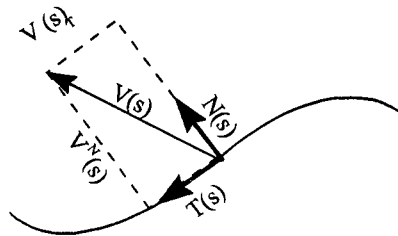


Figure 2 : décomposition du vecteur vitesse
(redessinée d'après E. Hildreth, *Proc. R. Soc. Lond. B*, 221, pp 189-220, 1984)

Considérons le cas de l'estimation du mouvement sur un contour. Soit $V(s)$, le vecteur de vitesse local le long du contour. $V(s)$ peut être décomposé en une composante tangentielle et une composante normale à la courbe comme le montre la Figure 2. $T(s)$ et $N(s)$ représentent les vecteurs unitaires dans les directions tangentielles et normales à la courbe, et $V^T(s)$ et $V^N(s)$ les deux composantes :

$$V(s) = V^T(s) \cdot T(s) + V^N(s) \cdot N(s) \quad (2.6)$$

Cependant, seule la composante normale $V^N(s)$ du vecteur vitesse est évaluée. La composante tangentielle $V^T(s)$ demeure "invisible" à ces mesures. L'évaluation du champ global des vitesses $V(s)$ à partir des données brutes de l'image est indéterminée. Mathématiquement, ceci s'explique par le fait que l'opérateur K défini par :

$$V(s) \cdot N(s) = K \cdot V(s) \quad (2.7)$$

n'est pas injectif.

De manière intuitive, la série de mesures données par $V^N(s)$ le long d'un contour devrait fournir une indication importante sur la nature du mouvement du contour. Toutefois, une contrainte générique supplémentaire est requise pour pouvoir déterminer l'unicité du mouvement. Par exemple, l'hypothèse d'un mouvement rigide dans le plan

image est suffisante pour estimer $V(s)$ de manière unique. Cette contrainte est très restrictive car elle ne couvre pas le cas des mouvements d'objets rigides dans l'espace. E.C. Hildreth [Hildreth, 1984a] a suggéré, en s'appuyant sur les travaux de B.K.P. Horn et B.G. Schunk [Horn & Schunk, 1981], d'employer une *contrainte de lissage* plus générale sur le champ des vitesses. Le choix de cette contrainte est guidé par des considérations mathématiques, l'unicité de la solution, et des hypothèses sur la physique. Les hypothèses physiques retenues correspondent dans ce cas aux déplacements rigides d'objets solides. Les surfaces sont alors généralement continues en rapport à leur distance avec l'observateur. Une surface compacte en mouvement doit générer en principe un champ de vitesse uniforme. Autrement dit, la variation des vecteurs vitesses le long d'un contour doit être la plus faible possible.

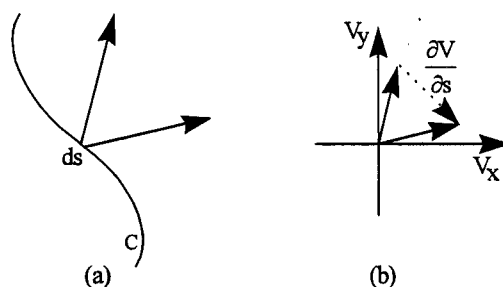


Figure 3 : mesure de variation dans les vitesses
(redessinée d'après E. Hildreth, *Proc. R. Soc. Lond. B*, 221, pp 189-220, 1984)

La variation locale de $V(s)$ le long du contour sur une longueur ds est donnée par la dérivée curviligne $\frac{\partial V}{\partial s}$. Dans la Figure 3.a, deux vecteurs proches le long du contour C sont montrés. Reportés dans l'espace des vitesses, Figure 3.b, le *vecteur variation* $\frac{\partial V}{\partial s}$ y est représenté. La mesure des variations totales des vitesses sur le contour complet se déduit en intégrant les variations ponctuelles afin d'obtenir la contrainte variationnelle suivante :

$$\|P \cdot V\|^2 = \int \left(\frac{\partial V}{\partial s} \right)^2 \cdot ds \quad (2.8)$$

où P est l'opérateur régularisant.

De façon plus général, s'il existe des erreurs dans les mesures de $V^\perp(s)$, l'alternative est de trouver $V(s)$ qui minimise :

$$\iint \left[(V \cdot N - V^\perp)^2 + \alpha \cdot \left(\left(\frac{\partial}{\partial s} V \right)^2 \right) \right] \cdot ds \quad (2.9)$$

On peut noter que ces approches correspondent respectivement à la deuxième méthode (relation (2.3)) et la troisième méthode (relation (2.4)) de régularisation vue au paragraphe précédent. L'unicité des solutions, démontré par E.C. Hildreth [Hildreth, 1984a] dans le cas de la première approche, est une conséquence des théorèmes standards de la théorie mathématique de la régularisation.

2.4 Optimisation

L'objet principal de l'optimisation est la construction d'algorithmes capables d'approcher au mieux une solution d'un problème de la forme : trouver z tel que

$$z \in Z, \text{ et } E(z) = \inf_{u \in Z} E(u) \quad (2.10)$$

Autrement dit, il s'agit de trouver un élément z de l'ensemble Z qui rend minimum la fonction E , communément appelée *fonctionnelle* en optimisation.

Dans les problèmes d'optimisation, il est nécessaire de s'intéresser aux deux points suivants :

- espace des solutions
- unicité de la solution si elle existe

L'unicité d'une solution éventuelle peut s'établir indépendamment de son existence, et le plus souvent à partir de la convexité de l'ensemble Z et de la stricte convexité de la fonctionnelle.

Deux types de problèmes d'optimisation sont à distinguer :

- les problèmes sans contraintes
- les problèmes sous contraintes

Un panorama mathématique complet sur les théories de l'optimisation est disponible dans [Cea, 1971] [Ciarlet, 1982] [Aubin, Nepomiaschty, & Charles, 1982].

2.4.1 Les problèmes d'optimisation sans contraintes

Un problème sans contrainte défini par une fonctionnelle E peut s'écrire :

$$\text{minimiser l'équation } E(\bar{z}) \quad (2.11)$$

La résolution de tel problème emploie des méthodes itératives, où partant d'un vecteur initial z_0 arbitraire, on construit une suite de vecteurs z_k , $k \geq 0$. L'équation d'Euler ou les inéquations d'Euler permettent de caractériser une solution éventuelle au problème [Ciarlet, 1982]. L'objectif des algorithmes est la convergence vers une solution du problème. Parmi les méthodes les plus connues, on peut citer *la relaxation, les méthodes de descentes de gradient (à pas optimal, à pas fixe, à pas variables, conjugué)* [Ciarlet, 1982]. Il est à noter que lorsque la fonctionnelle E à minimiser est quadratique, il est démontré que la méthode de relaxation se comporte exactement comme la *méthode de Gauss-Seidel* pour la résolution d'un système linéaire du type $A \cdot z = b$.

2.4.2 Les problèmes d'optimisation sous contraintes

Les solutions des problèmes d'optimisation sous contraintes constituent un sous-ensemble des solutions des problèmes sans contraintes. Un problème d'optimisation sous contraintes s'exprime de la façon suivante :

$$\text{minimiser l'équation } E(\bar{z}) \text{ sous la contrainte } g(\bar{z}) \quad (2.12)$$

où \bar{z} est le vecteur d'état des variables du problème. Ce vecteur représente une position dans l'espace des solutions du problème. Durant le processus de convergence de l'algorithme, l'état doit être attiré vers le sous-espace $g(\bar{z}) = 0$ et y rechercher la plus petite valeur de $E(\bar{z})$. Beaucoup de méthodes ont été développées dans le passé : elles utilisent *les multiplicateurs de Lagrange* ou *les relations de Kuhn et Tucker* (dans le cas où les contraintes sont des inégalités) pour caractériser les solutions éventuelles. Les algorithmes ont toujours pour nom relaxation ou méthode du gradient, tous deux étendus à ce genre de problèmes. Ces méthodes ne sont pas générales [Ciarlet, 1982], et les problèmes sous

contraintes sont souvent traités par d'autres approches. C'est le cas notamment de la *méthode de pénalisation*.

Cette méthode revient à ramener le problème d'optimisation sous contrainte en un problème non contraint en le reformulant sur la règle suivante :

$$\text{minimiser } E_{\text{pénalisation}}(\bar{z}) = E(\bar{z}) + \alpha \cdot g(\bar{z}) \quad (2.13)$$

où α est un coefficient qui pondère l'importance relative de la contrainte. Ce type de codage peut être étendu sans difficultés à des problèmes à contraintes multiples en créant autant de canaux de convergence qu'il y a de contraintes. Concrètement, le problème :

$$\text{minimiser l'équation } E(\bar{z}) \text{ sous les contraintes } g_i(\bar{z}) = 0; i = 1, \dots, n \quad (2.14)$$

est transformé dans le problème :

$$\text{minimiser } E_{\text{pénalisation}}(\bar{z}) = E(\bar{z}) + \sum_{i=1}^n a_i \cdot g_i(\bar{z}) \quad (2.15)$$

On peut remarquer que la méthode de régularisation (2.4) peut être vue comme un cas particulier de la méthode de pénalisation (2.13).

Une méthodologie de minimisation de fonctionnelles fondées sur l'exemple de $E_{\text{pénalisation}}$ (équation 2.15) issue de l'analyse canonique en mécanique statistique existe et est radicalement différente des approches déterministes conventionnelles, préalablement citées. Dans cette approche, on établit une analogie entre le problème à résoudre et un système physique tel que les verres de spin évoluant à une certaine température. Le principe est de simuler, à une température T , l'évolution d'un système physique jusqu'à l'équilibre thermodynamique, qui correspond à sa *thermalisation*. Des méthodes utilisant cette technique sont :

- le recuit simulé
- l'approximation en champ moyen
- le recuit en champ moyen.

Nous nous contenterons ici d'exposer le principe de la méthode du recuit simulé et renvoyons les lecteurs à [Stanley, 1971] pour l'approximation en champ moyen et pour le recuit champ moyen, à [Bilbro & Snyder, 1988] [Bout & Miller, 1989].

2.4.3 Le recuit simulé

Une bonne manière de trouver la solution optimale pour une fonctionnelle quelconque et même non convexe, l'état de plus basse énergie, d'un problème complexe est de le réchauffer à une température élevée avant de le refroidir lentement. Dans le cas des métaux, par exemple, on porte le corps à la température de fusion (ce qui correspond à une forte température) avant de le laisser refroidir en le plaçant dans des étuves à températures décroissantes bien contrôlées. On atteint ainsi des états cristallins de très faible énergie. Ce processus, appelé *recuit*, force l'état du système dans des régions à faible énergie tout en lui évitant d'être piégé dans des états correspondant à des minima locaux d'énergie élevée. La technique opposée est celle de la *trempe*, qui consiste à abaisser très rapidement la température du matériau en fusion. Dans ce cas, la structure du matériau est amorphe et l'état métastable qui s'ensuit correspond à un minimum local de l'énergie.

L'algorithme du recuit simulé a été introduit en 1983 par S. Kirkpatrick, C. Gelatt et M.P. Vecchi [Kirkpatrick, Gelatt & Vecchi, 1983] [Kirkpatrick, 1984]. L'idée de base consiste à considérer le système à optimiser comme un système physique défini par ses degrés de liberté $\{z_1, z_2, \dots, z_N\}$ et par son énergie globale $E(z) = \|Az - b\|^2 + \alpha \cdot \|Pz\|$ (cf relation 2.4). Le minimum absolu de cette énergie est atteint en simulant un processus de recuit d'une température élevée jusqu'à une température voisine de 0. L'algorithme peut alors se résumer de la façon suivante :

1. Commencer avec une température T_0 élevée
2. Thermaliser le système à cette température
3. Diminuer la température
4. Répéter les étapes 2 et 3 jusqu'à ce que le système ait atteint le minimum d'énergie.

La Figure 4 ci-dessous illustre la convergence de l'algorithme en visualisant le paysage énergétique.

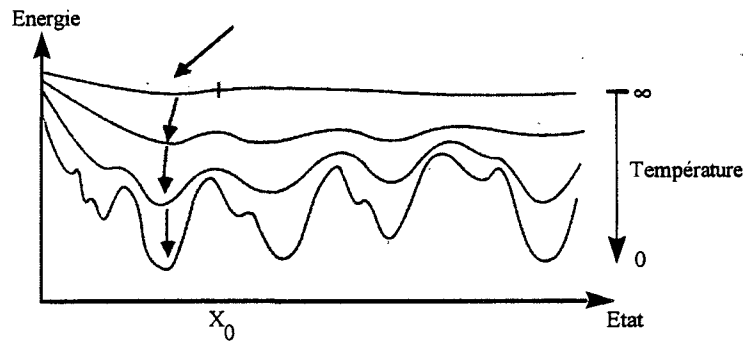


Figure 4 : méthode du recuit

◆ Phase de thermalisation (étape 2)

Elle est fondée sur les travaux de Metropolis [Metropolis & al, 1953] datant des années 50 qui proposa un algorithme de type *Monte Carlo* pour engendrer une séquence d'états convergeant à la limite vers une *distribution de Boltzmann*. Cet algorithme est le suivant :

1. Débuter avec une configuration initiale $\bar{z} = \{z_1, z_2, \dots, z_N\}$ aléatoire
2. Effectuer un léger déplacement (une transformation élémentaire) à partir de l'état précédent, par exemple en modifiant la valeur d'un seul degré de liberté z_j choisi aléatoirement pour aboutir à un nouvel état \bar{z}'
3. Evaluer la modification énergétique due à ce déplacement :

$$\Delta E = E(\bar{z}') - E(\bar{z})$$

où $E(\bar{z}')$ et $E(\bar{z})$ caractérisent respectivement les énergies pour les configurations \bar{z}' et \bar{z} .

4. si $\Delta E < 0$ (cas où le déplacement place le système dans un état d'énergie plus faible) alors le déplacement est accepté
5. si $\Delta E \geq 0$, le déplacement sera accepté avec la probabilité $\exp(-\Delta E/kT)$. En pratique, on tire un nombre aléatoire dans $[0, 1]$ et on le compare à $\exp(-\Delta E/kT)$. Si ce nombre aléatoire est inférieur à $\exp(-\Delta E/kT)$, la transformation est acceptée; sinon, elle n'est pas validée.

6. Répéter les étapes 2 à 5 jusqu'à ce que le système ait atteint l'équilibre, qu'il soit thermalisé.

La Figure 5 ci-dessous visualise la courbe des probabilités d'acceptation d'un déplacement en fonction de ΔE pour différentes valeurs de température.

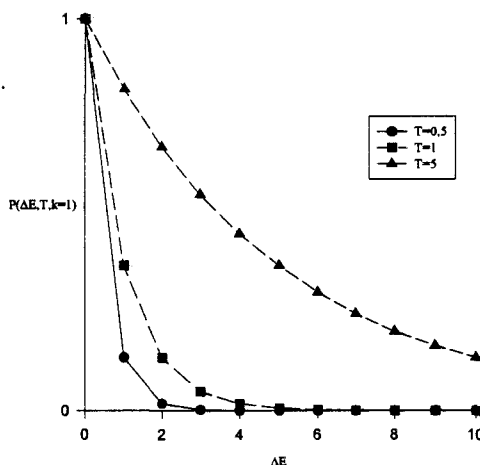


Figure 5 : Probabilité d'acceptation d'un déplacement

◆ Procédure du recuit (étape 3)

Des théorèmes de convergence asymptotique ont été établis et montrent que l'on peut s'approcher au plus près de la solution optimale si on est prêt à payer le prix nécessaire, des temps de calcul importants [Geman & Geman, 1984] [Gidas, 1985] [Laarhoven, 1987]. Ils prouvent que la loi de décroissance de la température doit être exponentielle. Toutefois, cette condition est très coûteuse en temps de calcul. De ce fait, on a souvent recours à certain compromis pour faire décroître la température.

Les solutions produites par l'algorithme du recuit simulé sont excellentes. Néanmoins, cette approche nécessite une importante quantité de tests et de calculs, due à sa nature intrinsèquement stochastique.

Les années 80 ont vu l'apparition de nouveaux outils, *les réseaux neuromimétiques*, issus des recherches menés sur la structure et le fonctionnement du cerveau humain dans le

but de déterminer des concepts et des principes fondamentaux qui pourraient être appliqués notamment aux problèmes d'optimisation [Sejnowski, 1986] [Cric & Asanuma, 1986]. Parmi les différents modèles de réseau existants, le réseau *récuratif* en particulier le *réseau de Hopfield* [Hopfield & Tank, 1985] offre un comportement dynamique gouverné par une fonction d'énergie décroissante le long d'une trajectoire, ce qui implique que le réseau converge vers un minimum local de cette fonction. Il en résulte donc que ce modèle de réseau est capable de résoudre des problèmes d'optimisation de la forme de l'équation (2.4).

2.5 Conclusion

Un grand nombre de problèmes de vision (ou de traitements d'images) peuvent s'exprimer sous la forme d'une fonction d'énergie. Pour la majorité d'entre-eux, ces problèmes sont mal posés au sens d'Hadamard car ils ne possèdent pas de solutions uniques. T. Poggio propose alors une méthode théorique satisfaisante pour rendre ces problèmes de vision primaire mal posés, correctement posés, *la régularisation*. Cette transcription du problème (équation 2.4) possède plusieurs caractéristiques très intéressantes :

- Elle est très simple à mettre en oeuvre.
- Elle converge de façon globale vers une solution exacte quand α tend vers l'infini [Hest, 1975].
- Elle permet de relaxer sans difficulté les contraintes en jouant sur les coefficients α .

De nombreux algorithmes d'optimisation, issus des mathématiques et de la physique statistique, existent et ont été employés pour résoudre ce type de problème. L'équation (2.4) peut aussi être considéré comme l'énergie d'un système physique complexe tel qu'un réseau de neurones récuratifs. Cette méthodologie de minimisation est alors radicalement différente des approches conventionnelles.

Chapitre 3

La Neuromimétique

Au cours de ce chapitre, nous dressons un bref tableau de la neuromimétique, concept connexionniste né de la fusion de deux sciences jusqu'à présent indépendantes l'une de l'autre : l'Informatique, sous l'aspect théorie des automates, et la Biologie, en particulier l'étude du cerveau. Deux grandes approches de la neuromimétique sont à distinguer. D'une part, celle qui fournit des modèles idéalisés grâce auxquels il est permis d'étudier le fonctionnement et les propriétés des systèmes nerveux biologiques. D'autre part, l'approche qui consiste à développer des réseaux, inspirés au départ par les connaissances biologiques, et dont le but est de réaliser des tâches technologiques bien précises. C'est sur cette dernière que porte principalement ce chapitre.

Afin de cerner les différentes structures de réseaux de neurones, nous allons les décomposer en trois niveaux. L'unité élémentaire de tous ces réseaux, le *neurone* (plus couramment désigné par le terme de *neurone formel*) fait l'objet de notre première étude. Puis nous sommes amenés à analyser la topologie des connexions reliant ces unités élémentaires dont l'importance est capitale puisqu'elle permet de classer les réseaux et de

définir le type de fonction qu'ils vont réaliser. Enfin, et ce avant de terminer par la dynamique des réseaux, nous évoquons succinctement les *réseaux hybrides* qui permettent quant à eux de réaliser des tâches plus complexes que les réseaux classiques ne peuvent accomplir.

On recense, aujourd'hui, un nombre important d'ouvrages traitant de la neuromimétique. Nous nous contentons simplement de citer [Davallo & Naim, 1990] [Rumelhart & McClelland, 1986] [Anderson & Rosenfeld, 1988] [Hecht-Nielsen, 1990] [Proceedings of I.E.E.E., 1990].

3.1 Historique

La neuromimétique est généralement perçue comme un domaine récent. Or, son développement remonte bien avant l'avènement des ordinateurs et a survécu à un revers majeur à la fin des années soixante. L'intérêt porté aux réseaux de neurones formels remonte à la première moitié de ce siècle et à l'article de deux cybernéticiens W.S. McCulloch et W. Pitts en 1943 dont les travaux sur des modèles physiques de réseaux constituèrent l'un des premiers jalons de ce domaine [McCulloch & Pitts, 1943]. Ils démontrèrent que leurs réseaux, constitués de neurones formels très simplifiés, pouvaient réaliser des fonctions logiques élémentaires.

Les premières simulations de systèmes neuronaux sur un ordinateur datent de 1954 et furent accomplies par B. Farley et W.A. Clark au sein du laboratoire de recherche d'I.B.M. [Farley & Clark, 1954], mais le premier réseau à remporter un certain succès est le *perceptron* de F. Rosenblatt [Rosenblatt, 1958] [Anderson & Rosenfeld, 1988]. L'engouement porté, à l'époque, à ce réseau était sa faculté, suite à un apprentissage, à classer de manière exclusive des modèles présentés en entrée en l'associant à une unité de sortie (un neurone). Le Perceptron était composé de trois couches dont l'intermédiaire est connue sous le nom de *couche d'association*. Il fit, par la suite, l'objet de nombreuses études [Rosenblatt, 1961] [Block, 1962] qui prouvèrent ses capacités à discriminer lors de son apprentissage des modèles linéairement séparables.

Parallèlement, d'autres modèles étaient étudiés, en particulier le réseau d'ADALINE (ADaptive LINear Element) de B. Widrow et M.E. Hoff [Widrow & Hoff, 1960] qui employait une procédure d'apprentissage plus sophistiquée que celle du Perceptron : la règle d'apprentissage par les moindres carrés, ou plus communément appelée règle de

Widrow-Hoff [Rumelhart & McClelland, 1986]. Cette règle était basée sur la minimisation itérative de l'erreur quadratique entre la sortie analogique désirée et la sortie analogique observée. Beaucoup d'autres méthodes ont été proposés à la même époque et publiés dans de nombreux actes de conférences [Cherry, 1961].

Néanmoins, ces différents modèles employaient généralement le neurone formel de McCulloch et Pitts, modèle de neurone simplifié. Ce neurone formel effectue une somme pondérée de ses entrées. Comparée à un seuil, cette somme va rendre le neurone actif (la sortie est égale à 1) si elle lui est supérieure, ou inactif (la sortie est égale à 0 ou -1) si elle lui est inférieure. Ce neurone formel, plus communément nommé automate à seuil, offre un potentiel de fonctions très limité. Les réseaux formés de ces neurones offraient une plus vaste classe de fonctions. Malheureusement, ils se restreignaient à la catégorie des fonctions linéairement séparables, puisqu'aucune règle d'apprentissage de l'époque ne s'adaptait au cas des réseaux de plus d'une couche. Ces limitations théoriques furent très vite comprises et démontrées mathématiquement dans l'ouvrage de Minsky et Papert [Minsky & Papert, 1969] qui provoqua (de manière malencontreuse) une certaine désaffection des chercheurs à ce domaine.

Malgré ce coup d'arrêt, plusieurs chercheurs ont persévéré, en renonçant à l'idée de créer "des machines intelligentes", et en s'attachant à développer des systèmes neuronaux pour résoudre des problèmes tels que la reconnaissance des formes. J. Anderson [Anderson, 1972], S.I. Amari [Amari, 1972], T. Kohonen [Kohonen, 1972] et S. Grossberg [Grossberg, 1976] furent parmi les premiers à travailler sur des modèles de réseaux à mémoire associative. De son côté, K. Fukushima [Fukushima, 1975] proposait un réseau à coopération/compétition basé sur le système visuel dénommé *Cognitron*. D'autres comme A.H. Klopff [Klopff, 1972] et P. Werbos [Werbos, 1974] s'attachaient à étudier des techniques d'apprentissage. Les progrès réalisés durant les années soixante dix et au début des années quatre-vingts sont à l'origine du renouveau de ce domaine. La popularisation de l'algorithme de rétro-propagation du gradient [Rumelhart, Hinton, & Williams, 1986] [Lecun, 1987], initialement publié par Werbos et réactualisé par D. Parker [Parker, 1982] [Parker, 1985], et les travaux de Hopfield sur les réseaux totalement interconnectés [Hopfield, 1982], sur lesquels nous revenons de manière plus détaillée au cours de cette thèse, sont sûrement les deux plus grandes avancées technologiques de la neuromimétique qui ont permis le renouveau de ce domaine.

Depuis, l'essor considérable qu'a connu ce domaine depuis une dizaine d'années a fait apparaître une grande variété de réseaux qui se distinguent entre-eux par l'application visée, la topologie et le type d'apprentissage utilisé. On trouve une classification détaillée de ces évolutions neuronales dans les références [Lippman, 1987] [Lemberg, 1988] [DARPA, 1988]. A titre indicatif, les Figure 1 et Figure 2 fournissent une illustration de l'évolution historique des principaux systèmes de réseaux de neurones. Enfin, l'émergence de la technologie VLSI apporte un autre intérêt de recherche à ce domaine concernant la réalisation de circuits neuronaux [Mead, 1989].

3.2 Le neurone formel

Le terme *neurone formel* est apparu pour la première fois en 1943 dans l'article de W.S. McCulloch et W. Pitts [McCulloch & Pitts, 1943] qui furent les premiers à proposer un modèle, très idéalisé, du neurone biologique. Le neurone y est décrit comme un automate possédant plusieurs entrées booléennes et une sortie également booléenne. Depuis cette époque, la forme du neurone s'est généralisée et l'on trouve divers modèles de neurones formels. Toutefois, sa structure générale reste identique et peut se résumer, de manière schématique, comme l'illustre la Figure 3. Ce neurone reçoit un signal ou une série de signaux en entrée issu d'autres neurones ou de données externes. Ces entrées, pondérées individuellement par des poids synaptiques T_{ij} qui représentent les forces de chacune des connexions, sont sommées et la valeur résultante est comparée au seuil θ_i pour fournir "le potentiel" ou encore "l'entrée totale" du neurone. La valeur de sortie ou "état" V_i du neurone se déduit alors du potentiel par une transformation mathématique à travers la fonction f . Cette fonction est appelée *fonction d'activation* ou encore *fonction de transfert*. Dans la suite, nous nous référons indifféremment à l'une ou l'autre de ces "appellations neuronales."

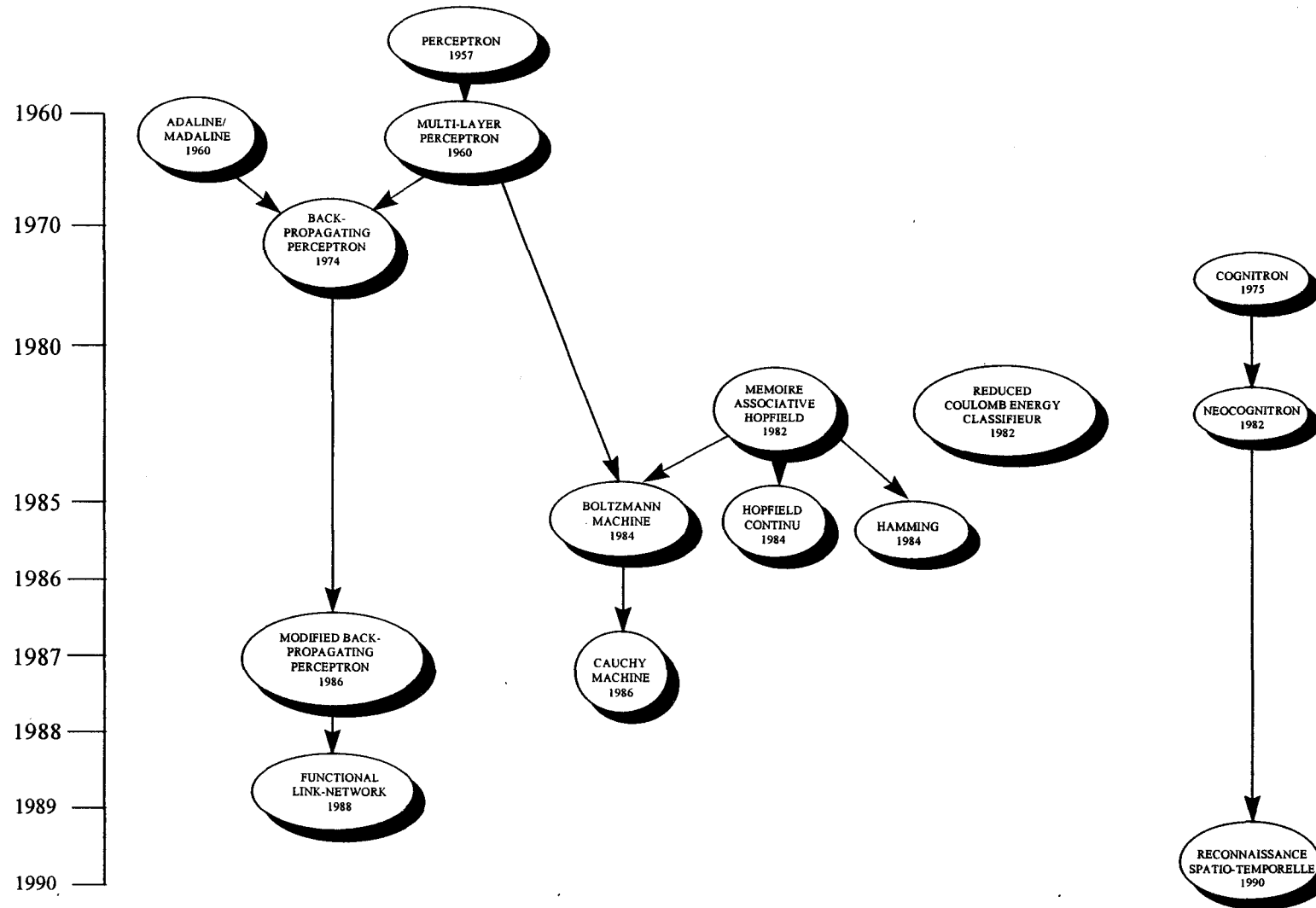


Figure 1 : Historique des réseaux de neurones

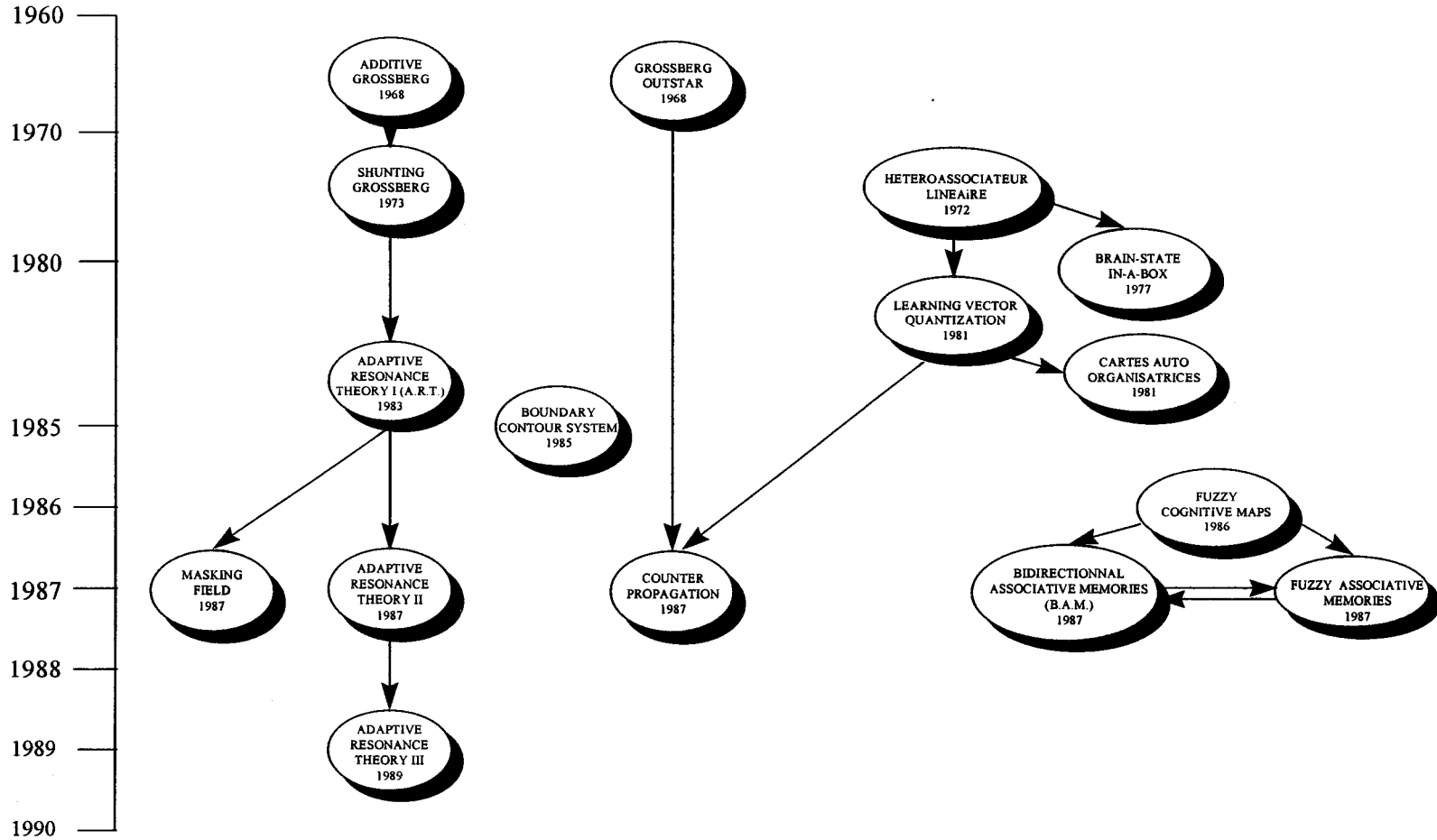


Figure 2 : Historique des réseaux de neurones

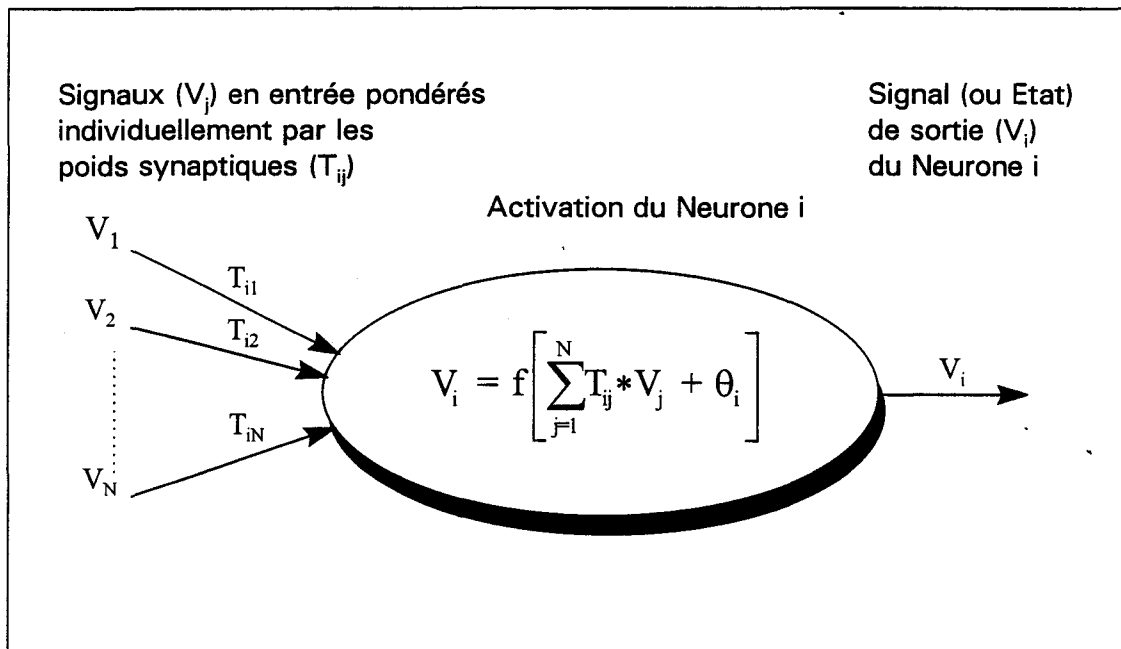


Figure 3 : Le neurone formel

Les différents choix de modèles de neurones formels se caractérisent par le type de fonction d'activation employée. On distingue ainsi trois grands modèles de neurones :

- le neurone formel binaire
- le neurone formel réel
- le neurone formel probabiliste

Enfin, il est à noter l'existence d'une autre famille de neurones formels. Nous ne nous étendons pas de manière détaillée sur les particularités de ces modèles et nous contentons simplement ici d'en énoncer l'idée sous-jacente. Celle-ci repose sur la généralisation de la forme de l'expression du potentiel en utilisant, par exemple, un polynôme de degré quelconque au lieu d'une simple combinaison linéaire. Ces unités élémentaires sont communément appelées *Cellules Sigma-Pi* [Koch & Ullman, 1985] [Durbin & Rumelhart, 1989], *Phi-machines* [Nilsson, 1965], ou encore *Fonctions Polynomiales Discriminantes* [Barron & Barron, 1988].

3.2.1 Neurone formel binaire

Le neurone formel binaire, aussi appelé automate à seuil, est donc le premier modèle développé par W.S. McCulloch et W. Pitts en 1943. Il possède N entrées

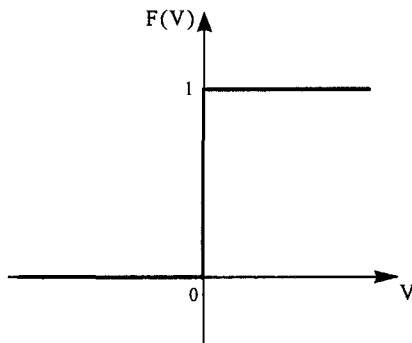
booléennes ou réelles (V_1, V_2, \dots, V_N) et une seule sortie booléenne. Si le potentiel du neurone u_i est positif, l'état V_i de l'automate devient actif et égal à +1; dans le cas contraire, il est inactif et égal à 0 :

$$V_i = 1 \quad \text{si} \quad \left(u_i = \sum_{j=1}^N T_{ij} \cdot V_j + \theta_i \right) > 0$$

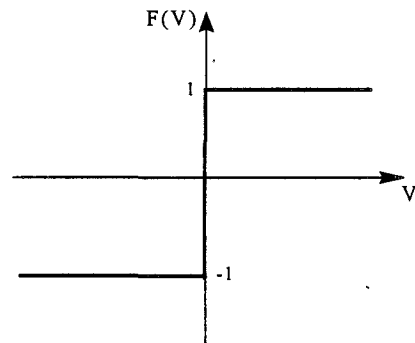
$$V_i = 0 \quad \text{sinon.} \tag{3.1}$$

La fonction d'activation utilisée est alors la fonction de Heaviside (Figure 4.a).

Il est possible, dans certaines situations, de trouver la valeur '-1' plutôt que '0' pour représenter l'inactivité du neurone. La fonction d'Heaviside est alors remplacée par la fonction Signe (Figure 4.b).



(a) fonction de Heaviside



(b) fonction Signe

Figure 4 : Fonctions d'activation binaires

Naturellement, d'un point de vue général, ces deux notations sont équivalentes puisqu'il est possible de passer du domaine $\{0, 1\}$ au domaine $\{-1, 1\}$ par un simple changement de variable ($V_i^{\text{Signe}} = 2V_i^{\text{Heavy}} - 1$). Ce changement se traduit alors par une modification de la valeur du seuil ($\theta_i^{\text{Signe}} = \theta_i^{\text{Heavy}} - 2 \sum_{j=1}^N T_{ij}$) du neurone.

Il est à noter que, dans certaines écritures, le seuil est considéré comme un poids particulier, $T_{i0} = \theta_i$, dont son entrée correspondante V_0 vaudrait toujours +1. L'équation du potentiel devient alors :

$$u_i = \sum_{j=0}^{j=N} T_{ij} \cdot V_j \tag{3.2}$$

3.2.2 Neurone formel à valeurs réelles

L'automate à seuil est intéressant lorsqu'il s'agit de traiter des discriminations, mais certaines applications nécessitent des fonctions plus complexes où l'emploi de neurones binaires devient inadéquat. Le modèle de neurone se généralise donc vers des sorties à valeurs réelles. Cet évolution se situe alors dans l'utilisation de fonctions d'activation continues à seuils. Ces fonctions sont généralement choisies croissantes et impaires (mais ceci ne constitue pas une restriction de principe). On trouve ainsi les fonctions linéaires à seuils (voir Figure 5) ou encore des fonctions sigmoïdes (voir Figure 6).

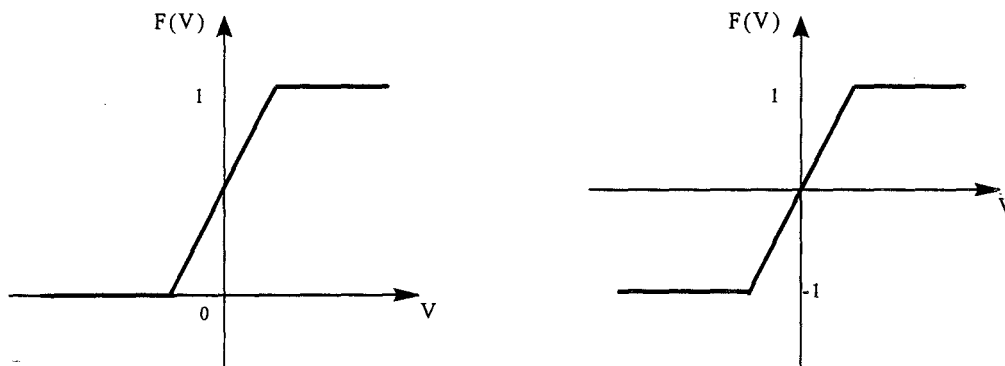


Figure 5 : Fonctions linéaires à seuils

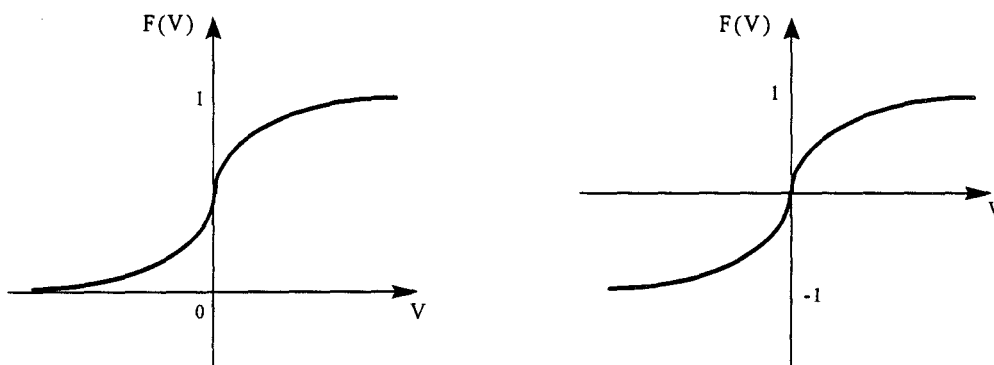


Figure 6 : Fonctions sigmoïdes

Ce dernier type de fonctions est apparu très intéressant en 1974 pour P. Werbos lorsqu'il proposa une nouvelle règle d'apprentissage basée sur la rétropropagation du gradient pour le perceptron multicouche [Werbos, 1974]. En effet, ces fonctions, au contraire des fonctions linéaires à seuils, sont dérivables en tout point. L'équation mathématique pour une fonction sigmoïde est de la forme suivante :

$$y = \frac{1}{(1 + e^{-\alpha \cdot x})} \quad (3.3)$$

où α est une constante qui définit la pente de la sigmoïde. Par exemple, la fonction arctangente est très souvent utilisée.

Enfin, un modèle bien moins commun, employé principalement dans le réseau *Radial Basis Function* [Powell, 1985] [Hush & Horne, 1993], a pour fonction d'activation une gaussienne (Figure 7). Ces neurones, situés sur la couche cachée, offrent une meilleure localisation des données présentées en leurs entrées [Hartman, Keeler, & Kowalski, 1990] [Lee & Kil, 1991].

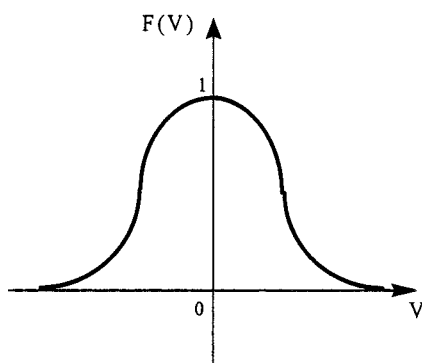


Figure 7 : fonction gaussienne

3.2.3 Neurone formel probabiliste

Ce dernier neurone formel est une évolution du modèle binaire, vu précédemment, qui offre la particularité d'introduire une probabilité de transition vers les différents états possibles au moyen d'une fonction d'activation stochastique. L'état V_i prend la valeur 1 si la probabilité $P(V_i=1) > P(V_i=0)$, et 0 dans le cas inverse. Les probabilités de l'état V_i étant données par :

$$P(V_i = 1) = \frac{1}{1 + \exp\left(-\frac{u_i}{T}\right)} \quad P(V_i = 0) = 1 - P(V_i = 1) \quad (3.4)$$

où T est un paramètre appelé "Température". Quand la température se rapproche de la valeur 0 (à basse température), la dynamique du neurone est proche du modèle binaire.

Le comportement de cet "automate probabiliste" est proche de celui des spins d'Ising en physique statistique.

Le neurone formel probabiliste est utilisé principalement dans les machines de Boltzmann [Hinton, Sejnowski, and Ackley, 1984] et de Cauchy [Szu, 1987].

3.3 La topologie des réseaux

La topologie d'un réseau caractérise l'organisation et la disposition de ces neurones en son sein. Les considérations topologiques sont particulièrement importantes puisqu'elles permettent de distinguer les différentes classes ou types d'architectures neuronales. Les diverses catégories de réseaux se particularisent au moyen des quatre critères suivants :

- Nombre de couches (monocouche, bicouche, ou multicouche)
- Nombre de neurones par couche
- Type de connexions (descendante, rétroactive, ou latérale)
- Degré de connectivité

En s'appuyant sur ces critères, nous pouvons identifier cinq grandes classes d'architecture neuronale :

- Réseau à couches
- Réseau récursif
- Réseau monocouche topologiquement ordonné (vector matching)
- Réseau bicouche rétroactif
- Réseau multicouche à coopération/compétition

auxquelles parfois vient s'en ajouter une sixième le *réseau hybride* pour la réalisation de tâche complexe qui nécessite l'interaction de plusieurs réseaux dit simples. Pour notre part, les réseaux hybrides sont développés séparément dans le paragraphe 3.4.

3.3.1 Réseau à couches

Dans les réseaux à couches (voir Figure 8), les signaux d'entrées se propagent à travers le réseau de couche en couche en direction de la sortie. Il n'existe aucune connexion entre les neurones d'une même couche ni même de connexions permettant un retour d'information d'une couche inférieure vers une couche supérieure. Cet architecture neuronal est typique au réseau appliqué à des problèmes de classification. L'exemple classique de cet architecture est le perceptron.

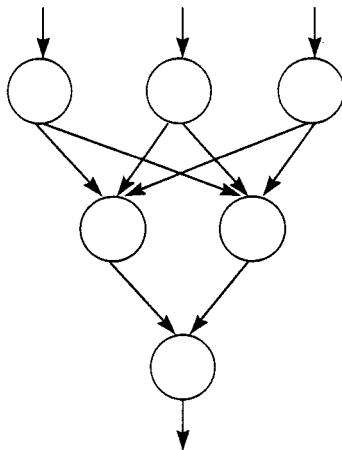


Figure 8: Réseau à couches

3.3.2 Réseau récursif

Cette seconde architecture (voir Figure 9) amplement connue utilise une seule couche de neurones entièrement ou partiellement connectés. Ces réseaux ont la capacité d'associer un motif, présenté en son entrée, à lui-même et le régénérer s'il est dégradé ou incomplet. On suppose pour cela que les différents motifs ont été préalablement mémorisés dans la structure interne du réseau. De ce fait, on surnomme aussi ces réseaux "réseau associatif" ou plus couramment "mémoire associative."

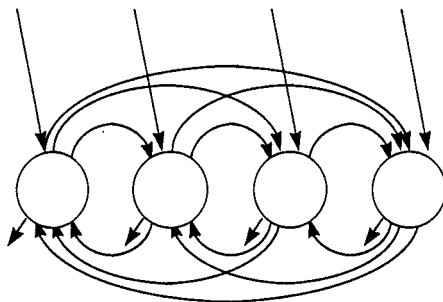


Figure 9 : Réseau récursif

Par la suite, Hopfield démontrera que, dans certaines conditions, le comportement dynamique des réseaux récursifs est gouverné par une fonction d'énergie décroissante le long d'une trajectoire, ce qui implique que ceux-ci convergent vers un minimum local de cette fonction. Ces réseaux peuvent donc s'employer, non seulement comme mémoire associative, mais aussi pour résoudre certains problèmes d'optimisation. Le Brain-State-in-

a-Box (BSB) d'Anderson [Anderson, Silverstein, Ritz, and Jones, 1977] et surtout le modèle d'Hopfield et Tank sont deux exemples de réseaux récurrents.

Nous revenons ultérieurement dans le chapitre 4 sur ces réseaux de manière plus détaillée.

3.3.3 Réseau bicouche rétroactif

Ce réseau à deux couches transmet les informations de l'entrée vers la sortie comme un réseau à couches classiques (voir §3.3.1) mais retourne également des données de la couche de sortie vers la première couche (voir Figure 10). Il existe donc des connexions de retour liant la couche de sortie à la première couche. Les poids de ces connexions sont différents de ceux liant les neurones de la première couche à ceux de la couche de sortie.

Ce type d'architecture est particulièrement efficace pour associer à un motif présenté en son entrée un autre motif à sa sortie. Nous parlons alors ici de "réseau hétéroassociatif."

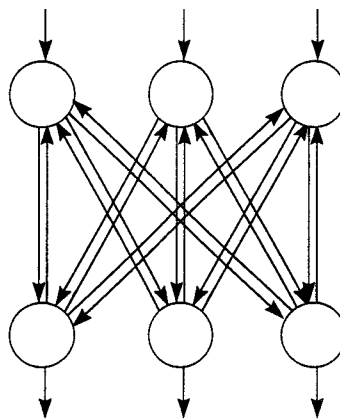


Figure 10 : Réseau bicouche rétroactif

Dans les plus récents des réseaux fondés sur cette topologie, une dynamique appelée *résonance* est introduite qui permet une *stimulation* des motifs sur la première couche et la couche de sortie jusqu'à ce qu'elles atteignent toutes les deux un état stable. Cette dynamique permet d'obtenir un meilleur rappel des motifs sur chaque couche. Deux des réseaux bicouches rétroactifs les plus populaires sont le réseau ART (Adaptive Resonance Theory) de Grossberg et Carpenter [Carpenter & Grossberg, 1987] et le BAM (Bidirectional Associative Memory) de Kosko [Kosko, 1987].

3.3.4 Réseau multicouche à compétition/coopération

Cette catégorie de réseau, au contraire des réseaux à couches (voir §3.3.1), possède en plus des connexions latérales qui lient les neurones d'une même couche (voir Figure 11). Les poids synaptiques excitateurs (à valeur positive) représentent des connexions de coopération, les inhibiteurs (à valeur négative) des connexions de compétition. Ce mode de conception trouve son inspiration dans certains réseaux biologiques. Par exemple, S. Grossberg et E. Mingolla [Grossberg & Mingolla, 1985] ont conçu le Boundary Contour System (BCS), réseau prévu pour la segmentation d'images, en s'appuyant sur les phénomènes observés lors d'études réalisées sur le système visuel.

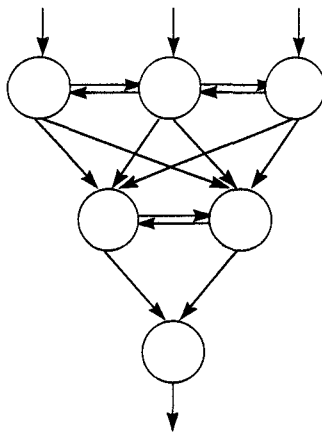


Figure 11 : Réseau multicouche à compétition/coopération

3.3.5 Réseau monocouche topologiquement ordonné

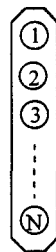


Figure 12 : Neurode

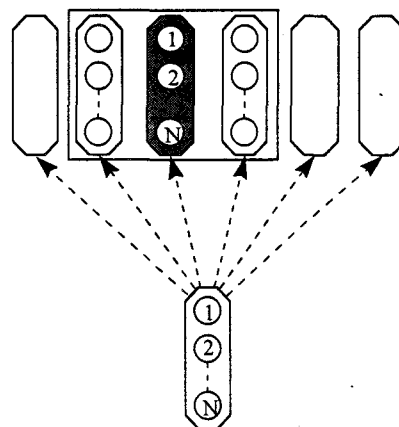


Figure 13 : Réseau monocouche topologiquement ordonné

Enfin, nous terminons par le réseau monocouche topologiquement ordonné. Ce choix est délibéré car ce dernier modèle se place à part dans le milieu de la neuromimétique. Afin de bien en comprendre les principes, il est nécessaire de faire abstraction des notions neuronales préalablement vues.

Cette architecture (voir Figure 13), unique en son genre, résulte principalement des travaux accomplis par T. Kohonen. Ce réseau organise, en effet, sa propre représentation des *catégories* en se servant directement des données fournies en entrée. Ces données sont symbolisées par un vecteur à N dimensions. Sa finalité est d'associer à un vecteur motif présenté en entrée un autre vecteur indiquant la catégorie à laquelle il appartient selon un critère de distance minimale (distance euclidienne par exemple).

Ce réseau ne possède donc pas de connexions explicites, comme ils en existent dans les modèles précédents, et le concept de neurone devient ici inadéquat. Ainsi, l'unité élémentaire (qui n'a plus aucun fondement neurophysiologique) se désigne sous le terme de *neurode* et se compose de N valeurs réelles correspondant à la dimension du vecteur, comme l'illustre la Figure 12.

Cette classe de réseau intègre le "Learning Vector Quantization (LVQ)" et les cartes autoorganisatrices, "Self-organizing Topology-Preserving Map (TPM)" développés tous deux par T. Kohonen [Kohonen, 1984].

3.4 Les réseaux hybrides

Nous venons de présenter les cinq grandes classes d'architectures neuronales classiques qui permettent de réaliser un grand nombre de tâches élémentaires. Dans le cadre de problèmes plus complexes, il est généralement nécessaire de recourir à un système combinant deux ou plusieurs réseaux communément appelé réseaux hybrides. Deux variétés de réseaux hybrides sont à distinguer :

- réseau fortement couplé
- réseau modulaire

Le réseau fortement couplé fusionne deux ou plusieurs architectures neuronales classiques. Il devient de ce fait impossible de les dissocier de manière individuelle. Il se comporte alors comme un réseau homogène capable d'accomplir des tâches plus

importantes que celles réalisées par les réseaux classiques. Le réseau de Hamming, un hybride du perceptron et du réseau *MAXNET* [Lipmann, 1987], ou encore le *counter-propagation network* [Hecht-Nielsen, 1987], croisement du réseau de Kohonen avec celui de Grossberg sont deux exemples de réseau fortement couplé.

L'autre approche consiste à concevoir des systèmes de réseaux individuels agissant l'un sur l'autre en s'appuyant sur les théories de G. Edelman [Edelman, 1987]. Un simple réseau classique accomplit, en général, une tâche à la fois. Or, de nombreux problèmes complexes requièrent de multiples étapes de traitement. La solution est de développer un *réseau modulaire*. L'idée est de décomposer le problème en différentes fonctionnalités et de leur associer à chacune un réseau spécifique (un module). Le couplage de ces divers modules doit suivre une certaine hiérarchie afin de répondre au problème posé.

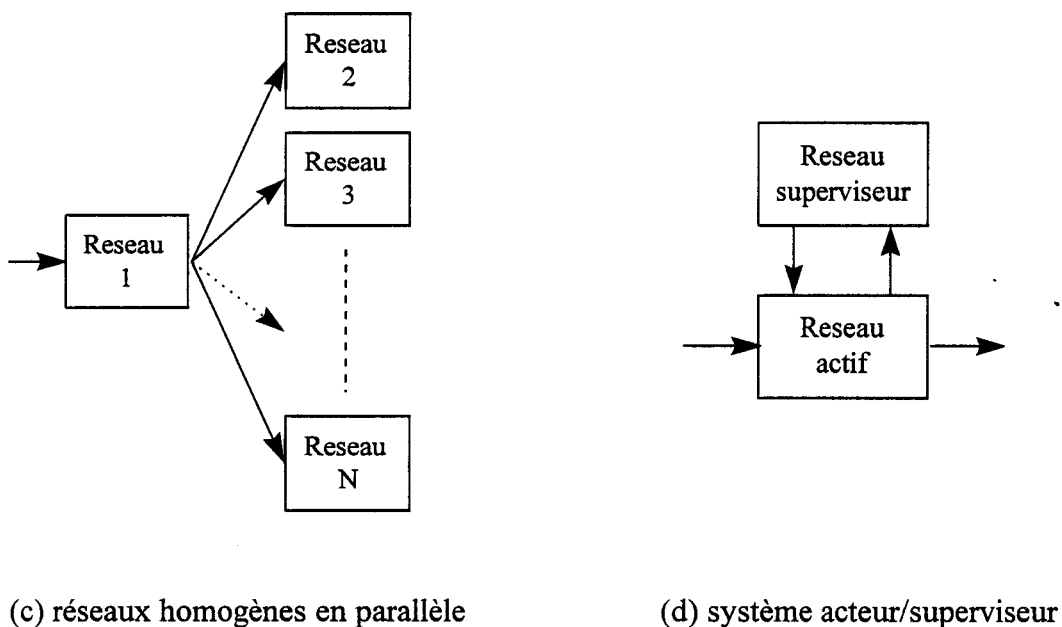
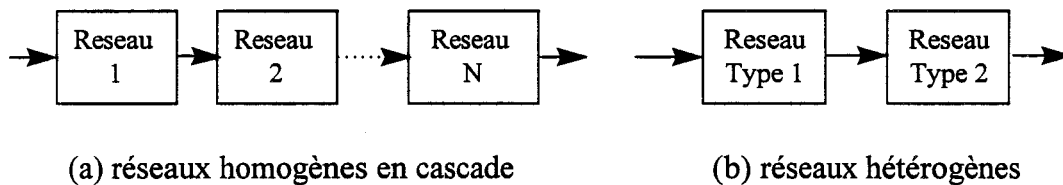


Figure 14 : organisations des réseaux modulaires

Il existe quatre types majeurs de couplage des réseaux (voir Figure 14). Le premier et le plus commun est de créer un système de réseaux similaires en les reliant hiérarchiquement comme l'illustre la Figure 14.a. Les différentes opérations sont alors

effectuées les unes à la suite des autres. La seconde méthode est de faire travailler les réseaux en parallèle comme le montre la Figure 14.c. Ceci est très utile lorsque plusieurs analyses sont à réaliser sur un même ensemble de données, ou lorsque des données doivent être opposées à plusieurs modèles différents.

La Figure 14.b présente un type de réseau modulaire construit avec des modèles de réseaux différents. Enfin, la Figure 14.d fait l'objet de recherches accrues. Ces systèmes, à l'intérieur desquels un réseau officie en tant que superviseur, sont très utiles dans des applications telles que la reconnaissance de formes avancée et le contrôle de processus.

Durant ces dernières années, la multiplication des réseaux hybrides s'est accentué.

3.5 La dynamique

Une fois fixés l'architecture et les modèles de neurones formels du réseau, les résultats que l'on peut en obtenir dépendent de la dynamique utilisée. Il en existe de plusieurs types, selon que les changements sont discrets ou continus dans le temps et selon l'ordre dans lequel s'effectuent ces changements.

3.5.1 Temps discret

Le temps est discrétisé et chaque pas de temps est marqué par un top d'horloge. On parle alors aussi d'itérations.

- *Mode synchrone*

Dans ce cas, tous les neurones changent d'états de façon synchrone en fonction des états à l'itération précédente :

$$V(t+1)=F(V(t)) \quad (3.5)$$

Ce mode de fonctionnement peut être comparé au schéma itératif de Jacobi pour la solution de systèmes linéaires [Golub & Van Loan, 1983].

- *Mode asynchrone*

Dans le mode asynchrone, les neurones du réseau sont mis à jour un à la fois dans un ordre fixé par une permutation sur l'ensemble (1, 2, ..., n). Dans cette optique, plusieurs variantes sont possibles dans l'ordre de passage des neurones, dont la plus courante est le *mode asynchrone séquentiel* où chaque neurone conserve la position qu'il occupe au sein

du vecteur d'état [Fogelman-Soulie & Weisbuch, 1987]. A chaque itération, un seul neurone change d'état en fonction des nouveaux états des k premiers neurones mis à jour et de l'ancien état des $(n+1-k)$ suivants :

$$\begin{aligned} \text{à } t & \quad V(t)=[V_1(t), V_2(t), \dots, V_k(t), V_{k+1}(t), \dots, V_n(t)] \\ \text{à } t+1 & \quad V(t+1)=[V_1(t), V_2(t), \dots, V_k(t+1), V_{k+1}(t), \dots, V_n(t)] \end{aligned} \quad (3.6)$$

Cette formulation mathématique rappelle l'itération de Gauss-Seidel en algèbre linéaire [Golub & Van Loan, 1983].

• Mode bloc-séquentiel

Soit $F=(f_1, f_2, f_3, \dots, f_n)$ la fonction de transition globale du réseau. On définit une partition ordonnée $Q = \{Q_1, Q_2, \dots, Q_p\}$ [Fogelman-Soulie, 1985] de l'ensemble $\{1, 2, \dots, n\}$ telle que :

$$\forall x \in Q_i, \forall y \in Q_j, i < j \Rightarrow x < y \quad (3.7)$$

Une itération bloc-séquentielle sur F associée à la partition ordonnée Q est alors définie comme suit. Pour tout $i \in Q_k$ ($k=1, 2, \dots, p$)

$$V_i(t+1) = f_i(z_j) \quad \text{où} \quad \begin{cases} z_j = V_j(t+1) & \text{si } j \in Q_1 \cup Q_2 \cup \dots \cup Q_{k-1} \\ z_j = V_j(t) & \text{sinon} \end{cases} \quad (3.8)$$

Ce mode est largement employé dans les modèles de réseaux à couches (cf §3.3.1). Pendant une itération, les neurones dont les indices appartiennent à l'ensemble Q_k calculent leurs états en fonction des nouveaux états ($V(t+1)$) des neurones dont leurs indices appartiennent à des ensembles d'indices inférieurs à k et des anciens ($V(t)$) pour les indices supérieurs à k . Tous les neurones dont les indices sont inclus dans l'ensemble Q_k changent leurs états synchroniquement.

On peut noter que le mode bloc-séquentiel représente la forme la plus générale de mise à jour puisque les deux autres en découlent comme cas particuliers. Pour $p=1$, c'est à dire lorsque $Q_1=(1, 2, \dots, n)$, on retrouve le mode synchrone, tandis que le cas $p=n$ correspond au mode asynchrone séquentiel.

3.5.2 Temps continu

Pour des changements d'états effectués en temps continu, les équations de changements d'états sont des équations différentielles. Citons celles proposées par Grossberg [Cohen & Grossberg, 1983]

$$\frac{dx_i}{dt} = a_i(x_i) \left[b_i(x_i) - \sum_{j=1}^N T_{ij} \cdot d_j(x_j) \right] \quad (3.9)$$

dont un cas particulier a été étudié par J. Hopfield [Hopfield, 1984] et F.J. Pineda [Pineda, 1987]

$$C_i \cdot \frac{du_i}{dt} = -\frac{u_i}{R_i} + \sum_{j=1}^N T_{ij} \cdot V_j + I_i \quad (3.10)$$

où C_i et R_i représente respectivement une capacité et une résistance.

Des résultats récents ont permis de caractériser le comportement de certains réseaux en fonction du mode d'itération qui y était appliqué, notamment quand une fonction d'énergie peut être attachée à celui-ci. Les travaux de Hopfield ont mis en valeur leurs capacités.

Une autre donnée, importante voire primordiale pour certain réseau tel que le perceptron, est la phase d'apprentissage. Cette étape a pour objectif d'adapter les valeurs des poids synaptiques. Cette apprentissage se fait de manière supervisée ou non supervisée. Ne s'inscrivant directement pas dans le cadre de cette thèse, cette partie est explicitée en Annexe I. En résumé, les différents algorithmes d'apprentissage développés peuvent être répertoriés sous deux grandes catégories.

3.6 Conclusion

Il est possible d'organiser le champ grandissant des réseaux de neurones en les envisageant à partir des trois niveaux de structure : le neurone, la topologie et les réseaux hybrides. Au niveau le plus bas (le neurone), la caractéristique, en général, la plus importante est la fonction de transfert dont Heaviside (ou signe), sigmoïde, et fonction linéaire à seuil sont les plus classiques.

Réseau	Année	Inventeur/ Concepteur	Applications principales	Avantages	Désavantages
ADALINE/ MADALINE	1960	B. Widrow	filtrage du signal adaptatif	Rapide, simple à implémenter et réalisation VLSI.	Relation linéaire entre entrée et sortie. Séparabilité linéaire.
Adaptive Resonance Theory (ART)	1983	G. Carpenter & S. Grossberg	Reconnaissance des formes	Apte à apprendre de nouveaux modèles et former de nouvelles catégories en plus de celles déjà apprises.	Inadéquate si modèle trop bruité.
Brain-state- in-a-box (BSB)	1977	J. Anderson	mémoire autoassociative	amélioration sensible des performances / réseau d'Hopfield.	Analyse partielle en termes de performance et de potentiel.
Bidirectional associative memory (BAM)	1987	B. Kosko	mémoire hétéroassociative	Architecture et apprentissage simples. Stabilité prouvée.	Faible capacité mémoire et faible exactitude de rappel.
Machine de Boltzmann et machine de Cauchy	1984 1986	G. Hinton, T. Sejnowski, D. Ackley H. Szu	Reconnaissance des formes (images, sonar, radar), optimisation	Représentation optimale des caractéristiques du modèle en atteignant le minimum de la fonction d'énergie.	Temps d'apprentissage très long pour la machine de Boltzmann, réduit dans le cas de la machine de Cauchy.

Tableau 1 : Applications des réseaux de neurones

Réseau	Année	Inventeurs	Applications principales	Avantages	Désavantages
Perceptron/ Perceptron multicouche	1958 1974-1986	F. Rosenblatt P. Werbos, D. Parker, D. Rumelhart	filtrage, reconnaissance de formes, classification, compression de données, segmentation, ...	Réalise de nombreuses opérations. Classifieur très étudié et employé avec beaucoup de succès	Temps d'apprentissage important.
Hopfield/Tank	1982-1984	J. Hopfield	Mémoire associative, Optimisation.	Concept simple, stabilité dynamique prouvée, facile à implémenter en VLSI.	Faible capacité mémoire. Possibilité d'états poubelles
Learning Vector Quantization	1981	T. Kohonen	Mémoire autoassociative Compression de données	auto-organisation des vecteurs	apprentissage lent
Cartes Auto-organisatrices	1981	T. Kohonen	Cartographie complexe Compression de données Optimisation, ...	auto-organisation dans les données.	Apprentissage lent
Cognitron/ Néocognitron	1975-1982	K. Fukushima	Reconnaissance de caractères manuscrits et autres.	Insensible aux phénomènes d'échelles, de rotation et de translation.	Nécessite plusieurs unités et couches de traitement, Structure complexe.
Radial Basis Function	1987-1988	Multiple chercheurs	Classification.	emploi de gaussienne sur couche cachée pour meilleur classification.	Méconnu.

Tableau 2 : Applications des réseaux de neurones

La topologie caractérise l'architecture des réseaux individuelles selon la connectivité des neurones en leur sein. Une classification des différents modèles de réseau est donc envisageable. Il existe ainsi cinq classes majeurs d'architectures : réseau à couches, réseau récursif, réseau monocouche topologiquement ordonné, réseau bicouche rétroactif, réseau multicouche à coopération/compétition.

Chacun des réseaux précédents est capable de réaliser une tâche différente. Les réseaux à couches sont souvent employés pour des travaux de classification. Les réseaux récursifs sont idéals comme mémoire auto-associative ou solveur de problèmes d'optimisation. Les réseaux bicouches rétroactifs sont très utiles pour associer un modèle à un autre. On parle alors de mémoire hétéro-associative. Enfin, les réseaux multicouches à coopération/compétition permettent d'effectuer des tâches plus complexes.

Les Tableau 1 et Tableau 2 présentent les différents domaines d'application des réseaux les plus courants, leurs avantages et leurs limitations.

Parfois, un simple réseau ne suffit pas pour résoudre certain problème. Dans ce cas, un système de plusieurs réseaux interagissants doit être créé. Ce sont les réseaux hybrides.

Le réseau conçu, les résultats que l'on peut en obtenir dépendent de la dynamique employée et des valeurs des connexions (poids synaptiques) qui lui ont été affectées lors de son apprentissage.

Chapitre 4

Les Réseaux récurrents

En 1982 paraissait aux Proceedings of The National Academy of Sciences of the U.S.A. un article de J. Hopfield intitulé : *Neural Networks and Physical Systems with Emergent Collective Computational Abilities* (Réseaux neuronaux et systèmes physiques possédant des capacités collectives émergentes de calcul). Cet article explicite a suscité, à son époque, un engouement explosif dans la communauté des physiciens et a, par contrecoup, réveillé l'intérêt des informaticiens. Il fut, ainsi qu'une poignée d'autres articles d'une qualité comparable, le point de départ du renouveau extraordinaire de ce champ d'activité scientifique : la Neuromimétique. L'approche relativement originale de Hopfield, comme en témoigne d'ailleurs la bibliographie, était la suivante :

- Partir des éléments très simplifiés du système nerveux connectés en réseau.
- Appliquer quelques idées de la physique statistique.
- Mettre en évidence des propriétés collectives comparables qualitativement à celles du cerveau.
- Proposer à partir de là de nouveaux circuits intégrés et de nouvelles architectures d'ordinateurs.

Un autre apport essentiel de cet article est l'analogie entre le modèle de réseau de neurones totalement connecté et certains cristaux aux propriétés étonnantes : *les verres de Spins*. Le comportement de l'un et de l'autre sont en effet décrits par une même *fonction d'énergie*. Cette analogie a permis d'utiliser un grand nombre de résultats issus de la physique statistique pour fournir une interprétation intéressante des réseaux récurrents mais aussi, pour pouvoir leur appliquer certaines techniques classiques de la mécanique statistique.

4.1 Mécanique statistique

La mécanique statistique résulte des travaux entamés à la fin du XIX^e siècle par L. Boltzmann et W. Gibbs, poursuivis par la suite par M. Planck, A. Einstein, P. Debye et bien d'autres. Son but est de décrire les propriétés des systèmes macroscopiques (c'est à dire à notre échelle) de certains matériaux à partir des propriétés microscopiques de leur constituants. Ces composants sont simplifiés au maximum de manière à pouvoir prendre en compte leur très grand nombre. Cette idée s'inscrit dans un courant d'idées récent, connu sous le nom de *physique des systèmes désordonnés* chez les physiciens. Le nombre gigantesque de constituants permet ainsi de se contenter de moyennes au lieu de rechercher toutes les configurations accessibles au système.

L'état microscopique d'un système physique peut être défini par une donnée de ' M ' coordonnées q_i et de ' M ' moments conjugués p_i ; par exemple, pour un système de ' N ' molécules, dans le cas où la structure interne de ces molécules peut être négligée, les q_i sont les ' $3N$ ' composantes cartésiennes des positions des molécules et les p_i sont les ' $3N$ ' composantes cartésiennes de leur quantités de mouvement. Un état microscopique peut donc être représenté par un point de coordonnées $q_1, q_2, \dots, q_M; p_1, p_2, \dots, p_M$ dans un espace abstrait à ' $2M$ ' dimensions, qu'on appelle l'espace des phases. Au cours du temps, le point représentatif de l'état du système évolue et décrit une trajectoire dans l'espace des phases.

Deux grandeurs fondamentales caractérisent les systèmes physiques :

- une fonction énergie
- la température

4.1.1 Energie

A chaque configuration du système peut être associée une fonction *énergie*. Son existence provient des symétries internes du système : les équations de la dynamique sont invariantes par translation du temps (on dit que les propriétés sont ergodiques) et les interactions entre particules obéissent à un principe d'égalité de l'action et de la réaction. Ces symétries n'existent que parce que les interactions sont prises en compte au niveau microscopique. En raison de l'analogie avec la physique, on se rend compte facilement que toute dynamique, dans des domaines très divers, évolue vers un attracteur et que ce dernier optimise une fonction de la configuration du système. Ces analogies vont de l'économie (maximisation des satisfactions), à l'évolution des espèces (maximisation de l'adaptabilité). En optimisation, le vecteur d'état \vec{z} du système défini dans la relation (2.12) du chapitre 2 est assimilé à une description de l'état microscopique d'un tel système qui évolue vers un attracteur correspondant à une solution du problème. L'état d'énergie minimal d'un tel système correspond à la solution optimale du problème d'optimisation.

4.1.2 Température

En général un système physique n'est pas isolé du reste du monde. Même s'il n'échange pas de matière avec l'extérieur, il peut y avoir des échanges d'énergie. Ce type d'interaction est pris en compte par la notion de *température*. Cet environnement de contact est supposé maintenu à une température connue 'T'; elle définit ce que l'on nomme un *thermostat*. L'énergie 'E' du système n'a donc pas une valeur fixée. Lorsque le système a atteint son équilibre, on montre que la probabilité pour qu'il occupe un état microscopique 'n' d'énergie 'E_n' est donnée par une loi de Boltzmann (ou de Gibbs) :

$$p(E_n) = \frac{\exp\left(-\frac{E_n}{kT}\right)}{Z(T)} \quad (4.1)$$

où k est la constante de Boltzmann et Z(T), une constante de normalisation appelée *fonction de partition*, qui a pour expression :

$$Z(T) = \sum_i \exp\left(-\frac{E_i}{kT}\right) \quad (4.2)$$

et qui traduit la somme de tous les états quantiques (ou toutes les configurations) accessibles au système.

Aux basses températures, les coefficients des exponentielles sont grands en valeur absolue et les probabilités des configurations de basses énergies sont donc très importantes par rapport à celles des configurations de haute énergie. Plus la température s'élève, plus les états d'énergie élevés interviennent.

A température nulle, le système évolue vers les minima d'énergie. A température non nulle, on retrouve un principe de minimisation où le système évolue de manière à minimiser une fonction, appelée *énergie libre*, donnée par :

$$F=E-T \cdot S \quad (4.3)$$

où 'E' est l'énergie interne, 'T' la température et 'S' l'entropie définie par :

$$S = \sum_n p(E_n) \cdot \log(p(E_n)) \quad (4.4)$$

Cette entropie est d'autant plus forte qu'un grand nombre d'états est accessible au système et son importance croît avec la température. En effet, aux basses températures, l'entropie intervient peu et les minima de l'énergie libre coïncident avec ceux de l'énergie interne. Au contraire, pour des températures élevées, la minimisation de l'énergie libre s'acquiert lorsque le système occupe, avec des probabilités non négligeables, un grand nombre de configurations, même si leur énergie est relativement élevée. En conséquence, les températures élevées favorisent le désordre.

4.2 Verres de spins

Les verres de spins constituent un modèle des verres en général, et des milieux magnétiques dilués en particulier qui offrent un parallèle frappant avec les réseaux récurrents. Une bonne vulgarisation est disponible dans [Hamman & Ocio, 1987]. Les solides peuvent exister sous deux formes extrêmes : la forme cristal, ordonnée, correspond à un arrangement régulier des atomes, tandis que la forme amorphe, désordonnée, est caractéristique des verres. L'arrangement irrégulier des atomes se traduit au plan macroscopique par des propriétés thermodynamiques différentes. Par exemple, la transition de phase liquide-solide, appelée vitreuse pour les amorphes, n'a pas le caractère abrupt que

l'on observe dans le cas des cristaux. La température de transition peut varier dans le cas d'un refroidissement trop rapide, et les effets d'hystérésis sont nombreux. Plutôt que de travailler sur des verres dans lesquels le désordre de nature structurale est difficile à contrôler, les physiciens travaillent souvent sur un système modèle plus simple, les milieux magnétiques dilués, dont on peut contrôler la composition, et sur lequel un grand nombre de mesures physiques sont possibles. Le désordre porte alors sur les interactions magnétiques entre les constituants.

Beaucoup de théories ont été développées pour modéliser les verres de spins et expliquer leur comportement. Nous pouvons, en particulier, citer le modèle de S. Kirkpatrick et D. Sherrington [Kirkpatrick & Sherrington, 1978]. Dans ce modèle, chaque spin interagit avec tous les autres d'une manière fixe mais aléatoire, indépendamment d'une notion de distance. Cette simplification justifie *la méthode du champ moyen*, et permet le calcul des propriétés thermodynamiques par la méthode dite des *répliques*. Les résultats en sont :

- En phase haute température, le système a accès avec une probabilité finie aux diverses configurations.
- Aux basses températures apparaît une phase des verres de spins dont le nombre d'attracteurs varie comme l'exponentiel du nombre des spins. Il est de l'ordre de $\exp(0.2 \cdot N)$ [Kirkpatrick & Sherrington, 1978]; ce résultat est démontré à température nulle.

Si les atomes 'i' et 'j' ont les spins ' S_i ' et ' S_j ', l'énergie d'interaction entre eux s'écrit :

$$H_{ij} = J_{ij} \cdot S_i \cdot S_j \quad (4.5)$$

où ' $S_i \cdot S_j$ ' représente le produit euclidien. Etant donnée une configuration 'w' des spins, l'énergie totale du système est donnée par l'Hamiltonien :

$$H(w) = -\frac{1}{2} \cdot \sum_{i=1}^N \sum_{j=1}^N J_{ij} \cdot S_i \cdot S_j - \sum_{i=1}^N S_i \cdot h_i \cdot F \quad (4.6)$$

où un vecteur unitaire $F \in \mathcal{R}^3$ représente la direction d'un champ magnétique externe et h_i son amplitude, et J_{ij} la constante d'interaction entre les spins S_i et S_j .

Les propriétés dynamiques du milieu magnétique dépendent du choix des J_{ij} . Ils existent ainsi diverses connectivités : cellulaire, complète ou aléatoire. Les valeurs des J_{ij} peuvent être tous identiques, ou différentes. Dans le cas des verres de spins, les interactions sont complètement aléatoires; pour le modèle ferromagnétique, les interactions sont positives et identiques ($J_{ij}=J$). Les réseaux récurrents constituent un cas intermédiaire entre les deux cas précédents.

L'étude de l'Hamiltonien de l'équation (4.6) est centrale en physique statistique. Sa complexité a conduit à faire des simplifications. Parmi celles-ci, l'une d'elle consiste à remplacer les coordonnées tridimensionnelles ' S_i ' et le champ magnétique par des scalaires à valeurs binaires [-1, +1]. C'est le modèle des spins d'Ising [Ising, 1925]. L'équation (4.6) devient :

$$H(w) = -\frac{1}{2} \sum_i \sum_j J_{ij} \cdot S_i \cdot S_j - \sum_i h_i \cdot S_i \quad (4.7)$$



A température nulle, la dynamique des spins est déterministe. Les spins sont mis à jour par une itération séquentielle aléatoire, et chacun prend l'état qui minimise son énergie d'interaction (équation (4.5)) avec ses voisins. Les attracteurs sont alors les points fixes qui minimisent la fonction d'énergie.

Si la température est non nulle, le comportement du spin n'est plus déterministe. Il a tendance à "préférer" les états d'énergie minimum, mais "l'agitation thermique" peut l'envoyer vers des états d'énergie supérieure. La transition vers l'état S_i suit une loi de probabilité :

$$P(S_i = 1) = \frac{1}{\left(1 + \exp - \frac{\sum_j J_{ij} \cdot S_i \cdot S_j}{k \cdot T} \right)} \quad (4.8)$$

Autrement dit, la prise en compte de la température dans un système dynamique se traduit par le caractère probabiliste des changements d'états de ses composantes. Si la somme des $J_{ij} \cdot S_j$ est fortement positive (c'est à dire en valeur absolue plus grande par rapport à $k \cdot T$), le spin S_i prend la valeur 1 avec une forte probabilité (proche de 1). Lorsque cette même somme est fortement négative, cette probabilité est quasi nulle et le spin prend

donc la valeur -1. Dans la zone indécise $4 \cdot k \cdot T$, le spin prend l'une ou l'autre valeur. Ce type de comportement est similaire au neurone formel probabiliste décrit au chapitre 2 §2.2.3.

Dans les systèmes physiques que nous venons de présenter ou citer, la carte des énergies (ou *le paysage énergétique*) ne se réduit pas à une vallée unique avec un seul attracteur à énergie minimum, mais comprend de nombreuses vallées séparées par des barrières. Ce phénomène se retrouve dans les réseaux récurrents. La température devient alors un paramètre important qui contrôle les aléas de la dynamique. Aux basses températures, le système reste bloqué au fond d'une vallée en fonction des conditions initiales; lorsque la température s'élève, les excursions en dehors du fond des vallées deviennent possibles jusqu'à ce que toutes les configurations du système deviennent accessibles.

4.3 Les réseaux récurrents

C'est un système dynamique multidimensionnel non-linéaire à temps *discret*. Il y a une analogie évidente avec les systèmes non linéaires à temps *continu*, décrit par des équations différentielles ordinaires et dont un des problèmes concerne la stabilité des points d'équilibre. Ce problème s'étudie classiquement via une technique appelée seconde méthode de Lyapunov [Barnett & Storey, 1970].

4.3.1 Le modèle discret

4.3.1.1 Définitions

A tout moment, le réseau est caractérisé par un vecteur d'état discret. De façon plus précise, pour un réseau récurrent à 'N' neurones, le vecteur d'état à l'instant 't' est de la forme:

$$V(t) = [V_1(t), V_2(t), \dots, V_N(t)]^T \in [-1, 1]^N \quad (4.9)$$

Soit $\mathbf{T} = [T_{ij} : 1 \leq i, j \leq N]$ une matrice symétrique d'ordre N et $\mathbf{I} = [I_i : 1 \leq i \leq N]$ un N-vecteur réel. L'évolution du système au cours du temps est régie par une équation dynamique de la forme :

$$V_i(t+1) = \text{sgn} \left(\sum_{j=1}^N T_{ij} \cdot V_j(t) + I_i \right) \quad \text{avec } i=1, 2, \dots, N \quad (4.10)$$

La matrice T et le vecteur I traduisent les paramètres du réseau, respectivement les poids synaptiques et les entrées externes (ou seuils des neurones). On constate qu'il existe 2^N vecteurs d'état possibles (ou configurations du réseau) qui coïncident avec les sommets de l'hypercube C^N de dimension N . Durant son évolution temporelle, le réseau parcourt les sommets de l'hypercube et effectue donc une application de C^N sur lui-même. La Figure 1 représente les différents vecteurs d'état possibles pour un réseau de 3 neurones au sein d'un hypercube de dimension 3.

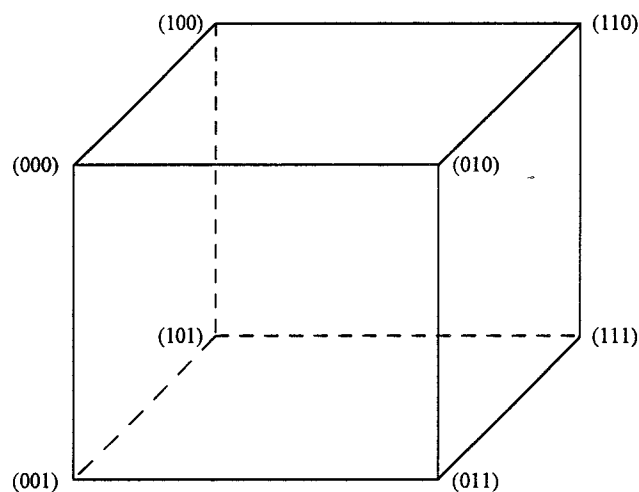


Figure 1 : Représentation cubique des configurations possibles pour un réseau de 3 neurones

Il est à noter que l'on peut trouver la fonction de Heaviside en remplacement de la fonction signe (sgn) comme fonction d'activation. Les états des neurones appartiennent alors à $[0, 1]$, mais le comportement du réseau reste inchangé (voir chapitre 2 §2.2.1).

Si l'on identifie l'état ' V_i ' d'un neurone formel à la sortie ' S_i ' du spin magnétique d'un atome, les poids synaptiques ' T_{ij} ' aux interactions ' J_{ij} ' et l'entrée externe ' I_i ' au champ magnétique externe ' h_i ', on retrouve les caractéristiques des milieux magnétiques dilués.

4.3.1.2 Dynamique

Au cours du temps, les vecteurs d'état successifs de l'équation (4.9), fournis par l'équation fondamentale (4.10), peuvent s'obtenir de différents façons :

- Mode synchrone
- Mode asynchrone
- Mode bloc séquentiel

comme cela a été décrit précédemment au chapitre 2 §2.5.1.1.

Le mode asynchrone est le plus souvent employé car il assure automatiquement la convergence du réseau vers un minimum local. J. Hopfield [Hopfield, 1982] applique cette dynamique à son modèle originel au moyen de la *méthode de Monte-Carlo* pour tirer au sort le neurone susceptible de changer d'état. Cette technique est très fréquemment employée dans la simulation numérique de certains systèmes physiques.

4.3.1.3 Energie

De par l'analogie avec les spins, le comportement de ce réseau peut se caractériser sous forme d'une énergie globale par :

$$E = -\frac{1}{2} \cdot \sum_i \sum_j T_{ij} \cdot V_i \cdot V_j - \sum_i I_i \cdot V_i \quad (4.11)$$

A T_{ij} et I_i fixés, l'énergie est fonction des états de chaque neurone. On définit ainsi un paysage énergétique discret à N dimensions et 2^N configurations comme l'illustre la Figure 2 dans le cas d'un réseau de 3 neurones. Les minima locaux de cette fonction (4.11) correspondent aux états stables du réseau.

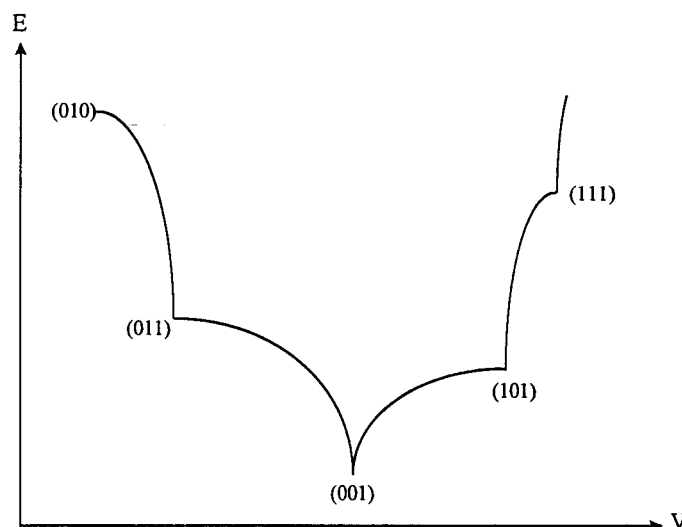


Figure 2 : Représentation de l'énergie en fonction du vecteur d'état V au voisinage d'un minimum local

Sous l'hypothèse d'une matrice à diagonale nulle, $T_{i,i}=0$, le réseau converge vers un état stable caractérisé par un minimum local de la fonction d'énergie quelque soit le mode

de mise à jour du réseau. Ce minimum dépend de la valeur initiale du vecteur d'état du réseau.

L'introduction de cette fonction d'énergie a permis d'utiliser un grand nombre de résultats de la physique statistique et a motivé les physiciens dans l'étude de ce modèle comme le montre le nombre important des travaux réalisés parmi lesquels nous pouvons, en particulier, citer [Amit & Gutfreund, 1985], [Amit, 1987] et [Peretto, 1987].

4.3.2 Modèle continu

Ce modèle est fondé sur des variables et des états à valeurs continues dans l'espace des états. Le comportement de ce réseau est similaire au modèle discret précédent. Les neurones sont à valeurs réelles (voir chapitre §2.2.2), et de ce fait plus proche du "neurone biologique." De plus, la réalisation d'un circuit électronique devient plausible. Dans ce modèle, le vecteur état est contraint à l'intérieur de l'hypercube C^N .

4.3.2.1 Dynamique

Les variables de sortie (ou états) V_i d'un neurone 'i' sont comprises dans un intervalle $[0,1]$ et continues. La fonction d'activation f est monotone croissante et il est nécessaire qu'une fonction inverse existe, c'est à dire f^{-1} définie. Le choix typique est la fonction sigmoïde (voir chapitre 2 §2.2.2) :

$$f(u_i) = \frac{1}{1 + \exp\left(-\frac{u_i}{u_0}\right)}$$

$$u_i = f^{-1}(V_i)$$
(4.16)

où u_0 détermine la pente de la sigmoïde. En faisant décroître ce paramètre pendant le processus de convergence du réseau, on garantit que les minima locaux de l'énergie correspondent effectivement aux sommets de l'hypercube C^N du réseau discret.

Pour la réalisation électronique de cette fonction, Hopfield [Hopfield, 1984] a proposé d'utiliser la relation entrée-sortie d'un amplificateur opérationnel non linéaire avec un temps de réponse négligeable. La simulation d'un circuit électronique du neurone formel devient donc possible. La Figure 3 montre le schéma du neurone électronique. De ce fait,

la dynamique continue en temps (et asynchrone) du réseau est régie par une équation différentielle :

$$\frac{du_i}{dt} = -\frac{u_i}{\tau} + \sum_{j=1}^N T_{ij} \cdot V_j + I_i$$

$$V_i = f(u_i) \quad (4.17)$$

où τ représente une constante caractéristique de temps du circuit ($\tau = R \cdot C$), T_{ij} , les poids synaptiques ($T_{ij} = 1/R_{ij}$), et f le gain de l'amplificateur.

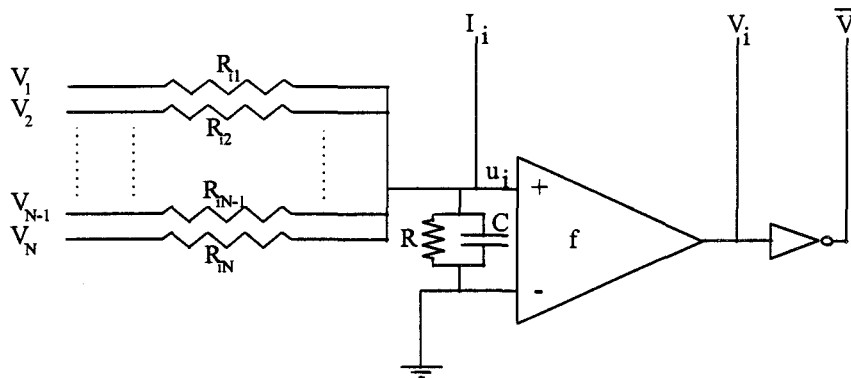


Figure 3 : Le neurone électronique

4.3.2.2 Energie

L'énergie E du réseau est définie par :

$$E = -\frac{1}{2} \cdot \sum_i \sum_j T_{ij} \cdot V_i \cdot V_j - \sum_i I_i \cdot V_i + \sum_i \frac{1}{R_i} \cdot \int_0^{V_i} f^{-1}(V) \cdot dV$$

$$(4.18)$$

avec
$$\frac{1}{R_i} = \sum_{j=1}^N \frac{1}{R_{ij}} + \frac{1}{R}$$

La dérivée temporelle, lorsque la matrice \mathbf{T} est symétrique, est :

$$\frac{dE}{dt} = -\sum_i \left(C \cdot \frac{\partial f^{-1}(V_i)}{\partial V_i} \cdot \left(\frac{dV_i}{dt} \right)^2 \right) \quad (4.19)$$

Puisque f^{-1} est une fonction monotone croissante et la valeur de la capacité C est positive, le produit à l'intérieur des parenthèses de l'équation ci-dessus est positif ou nul. En conséquence, le réseau se stabilisera dans une configuration à énergie minimale quand $\frac{dV_i}{dt} = 0$. La fonction d'énergie E est donc une fonction de Lyapunov.

Hopfield [Hopfield, 1984] démontra l'analogie de ce modèle avec un réseau à neurones binaires stochastiques.

4.3.3 Mémoires associatives

En 1982, J. Hopfield [Hopfield, 1982] suggère l'idée que, dans le cerveau humain, l'information circule et qu'une décision est prise par étapes successives. Selon lui, le système nerveux recherche des états stables, attracteurs dans son espace d'états. Les états voisins tendent à se rapprocher d'un état stable, ce qui autorise la correction des erreurs, et la capacité à compléter des informations manquantes. Ce comportement correspond à une *mémoire adressable par son contenu* (C.A.M.) : une forme mémorisée est retrouvée par une stabilisation du réseau, si ce dernier a été stimulé par une partie suffisamment proche de cette forme.

J. Hopfield propose ainsi un modèle capable de réaliser de telles propriétés, basé sur un réseau récurrent discret avec des neurones du type McCulloch & Pitts. Dans ce réseau, la connectivité est complète à l'exception des termes diagonaux T_{ii} (correspondant à la connexion du neurone sur lui-même), qui sont nuls. Bien entendu, il est possible que d'autres poids synaptiques soient eux aussi nuls. De plus, les T_{ij} sont symétriques ($T_{ij} = T_{ji}$). La matrice T est donc symétrique à diagonale nulle ce qui permet d'éviter les cycles et donc d'assurer sa convergence vers un état stable.

Le problème à résoudre consiste donc à imposer les formes à mémoriser comme points fixes ou attracteurs de ce réseau. Ceci se fait par un choix convenable de la matrice des poids synaptiques T . Cette phase, qui représente l'apprentissage non-supervisé du réseau (cf Annexe II), a, pour Hopfield, trouvé réponse grâce aux travaux de D. Hebb [Hebb, 1949]. Dans son ouvrage *The Organization of Behavior*, Hebb suggérait que l'apprentissage d'un comportement cognitif faisait intervenir au niveau cellulaire le renforcement des synapses des neurones dont l'activité est corrélée. La présentation

successive des formes à mémoriser va permettre d'adapter ces poids dont l'expression est donnée par :

$$T_{ij} = \frac{1}{M} \cdot \sum_{S=1}^M \mu_i^S \cdot \mu_j^S \quad (4.20)$$

où μ^S représente le vecteur prototype de la forme S et M le nombre de formes à mémoriser.

Différents résultats sous forme de simulations ont démontré les capacités d'un tel système. La Figure 4, extraite de l'article de R.P. Lipmann, présente un exemple de fonctionnement du réseau de Hopfield en tant que mémoire associative. Huit modèles prototypes (partie A) de dimensions 10×12 sont codés en -1 pour le blanc et 1 pour le noir. Le réseau constitué de 120 neurones apprend, au moyen de la règle de Hebb, à mémoriser ces exemples qui représentent donc ses attracteurs. Le chiffre "3", préalablement bruité, est ensuite appliqué à l'entrée du réseau à l'instant $t=0$. La partie B de la figure montre le résultat obtenu après 7 itérations. On constate que le réseau a convergé vers le modèle du chiffre "3" mémorisé.

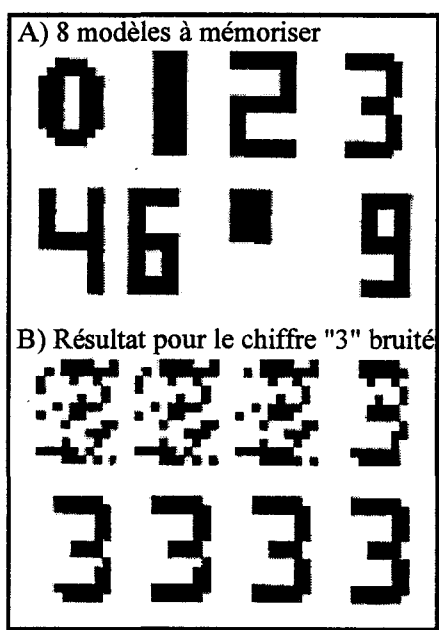


Figure 4 : Réseau de Hopfield comme mémoire associative
(D'après R.P. Lipmann, *I.E.E.E. A.S.S.P. Magazine*, pp 4-23, 1987)

Le réseau de Hopfield possède deux inconvénients majeurs en tant que mémoire associative. Premièrement, le nombre de modèles à mémoriser est limité. Si trop de modèles sont mémorisés, le réseau peut converger vers des "états poubelles" autrement dit vers des attracteurs différents de ceux qui caractérisent les modèles souhaités. Hopfield montra, de manière empirique, que la *capacité* du réseau (le nombre de modèles pouvant être mémorisés) ne devait pas excéder [Hopfield, 1982] :

$$M \leq 0.15 \cdot N \quad (4.21)$$

Cette capacité de mémorisation a fait l'objet de nombreuses études théoriques lorsque le nombre N de neurones devient très important [Abu-Mostafa & Saint Jacques, 1985] [McEliece, Postner, Rodenmich, & Venkatesh, 1987].

La seconde limitation du réseau est que les différents modèles à mémoriser soient suffisamment distincts les uns vis à vis des autres. En d'autres termes, ils doivent être orthogonaux sinon, la corrélation qui existe entre eux entraînera des défauts de mémorisation [Dembo, 1989]. Une variante de la loi de Hebb, *la loi de Hebb pondérée*, où on donne une pondération différente ' λ_s ' aux poids (équation (4.14)) de sorte à renforcer les modèles pas suffisamment bien mémorisés a été proposée [Amari, 1972].

$$T_{ij} = \frac{1}{M} \cdot \sum_{s=1}^M \lambda_s \cdot \mu_i^s \cdot \mu_j^s \quad (4.22)$$

Deux autres règles d'apprentissage, *la règle de projection ou du pseudo-inverse* [Amari, 1972] [Personnaz, Guyon, & Dreyfus, 1986] que l'on peut formellement rattacher à la règle de Widrow-Hoff, et *la règle du minover* qui reprend la règle d'apprentissage du perceptron, permettent de déterminer les valeurs des poids synaptiques. Nous renvoyons les lecteurs à l'ouvrage de T. Kohonen [Kohonen, 1984] pour d'amples informations sur ces méthodes.

Toutefois, la règle de Hebb demeure la plus courante car plus proche de la réalité biologique et surtout elle évite de recalculer complètement la matrice T des poids synaptiques lorsqu'on ajoute un modèle supplémentaire à mémoriser ce qui n'est pas le cas avec les deux autres règles. En effet, les nouvelles valeurs des poids T'_{ij} se déduisent de l'ancienne formulation (équation (4.20)) :

$$T'_{ij} = \frac{1}{M+1} \cdot (M \cdot T_{ij} + \mu_i^{M+1} \cdot \mu_j^{M+1}) \quad (4.23)$$

D'autres architectures neuronales parmi lesquelles on peut citer le *Brain-State-in-the-Box (B.S.B.)* d'Anderson [Anderson, Silverstein, Ritz, and Jones, 1977], les *Bidirectional Associative Memories (B.A.M.)* de Kosko [Kosko, 1987] offrent aussi des facultés de mémoires associatives. Dans ce dernier réseau, la mémoire n'est plus auto-associative mais hétéro-associative. Enfin, une des applications la plus connue du modèle de Hopfield comme mémoire associative est la machine de Boltzmann. L'architecture et l'algorithme d'apprentissage ont été développés par une équipe américaine composée de D.H. Ackley, G.E. Hinton et T.J. Sejnowsky [Ackley, Hinton & Sejnowsky, 1985]. Ce réseau introduit la notion du recuit simulé et un algorithme d'apprentissage à deux temps qui permet de placer les minima locaux là où on le souhaite. Toutefois, l'inconvénient majeur de cette machine reste un temps d'apprentissage très long, puisque la modification d'un poids nécessite l'estimation de probabilités. Cet algorithme se trouve de ce fait peu utilisé dans la pratique.

4.3.4 Réseau de Hopfield et Optimisation

La fonction d'énergie joue un rôle très important. Elle ne se limite pas à garantir l'absence de cycles pour les mémoires associatives, mais permet aussi de concevoir un réseau récurrent comme un outil technique permettant de trouver un optimum pour toute fonction de coût qui peut se mettre sous la forme d'une fonction d'énergie associée à un réseau de ce type. En 1985, dans son article intitulé *Neural Computation of decisions in Optimization problems*, J. Hopfield [Hopfield & Tank, 1985] introduit l'idée d'utiliser son modèle de réseau récurrent à la résolution de problèmes d'optimisation.

Dans l'approche précédente, les mémoires associatives présentées ci-dessus, les poids se déterminent à partir des états stables imposés (modèles à mémoriser). Quelle que soit la règle d'apprentissage employée, il était possible de prévoir la stabilisation du réseau en un point fixe. En effet, la matrice des connexions \mathbf{T} est définie symétrique à diagonale nulle et la théorie générale permet de prévoir que le réseau converge vers des états stables.

Pour les problèmes d'optimisation, les états ne sont pas connus à l'avance mais se caractérisent par une fonctionnelle d'énergie ou fonction coût. Le but de cette approche est

donc de déterminer les valeurs des connexions de sorte que les états stables du réseau minimisent la fonctionnelle d'énergie du problème à résoudre.

4.3.4.1 La méthodologie

De la façon la plus générale, la méthodologie proposée par Hopfield consiste à déterminer, en prenant en compte le problème à résoudre :

- la signification d'un automate qui représente une partie de l'information
- les états qu'il peut prendre, autrement dit le choix de sa fonction d'activation
- les connexions du réseau qui doivent :
 - d'une part représenter les contraintes réelles du problème
 - d'autre part assurer la convergence du réseau vers des états stables.

Pour cela, la matrice des connexions doit être symétrique.

- La fonction d'énergie associée au réseau
- la fonctionnelle d'énergie qui amène à résoudre le problème

La méthode consiste alors à identifier la fonction d'énergie du réseau à la fonctionnelle d'énergie à minimiser de façon à déterminer les paramètres du réseau : poids synaptiques et entrées externes.

Cette méthodologie a été expérimentée sur un problème classique, le problème du voyageur de commerce [Hopfield & Tank, 1985].

Ce problème s'exprime de façon très simple. On dispose de n villes A, B, C, ... dont on connaît les distances deux à deux d_{AB} , d_{AC} , Le but est de déterminer un parcours, passant une fois et une seule par chaque ville, en revenant à son point de départ et tel que la distance totale du parcours d soit minimale.

Pour caractériser le système, Hopfield [Hopfield & Tank, 1985] choisit de représenter le parcours par un tableau (villes, étapes) de dimension $n \times n$. Une case (X,i) est à 1 si la ville X est visitée à l'étape i, sinon elle est à 0. L'état V_{Xi} d'un neurone va traduire cette définition. Le réseau comporte donc $N=n^2$ neurones binaires.

La fonctionnelle d'énergie

$$E = \frac{A}{2} \cdot \sum_X \sum_i \sum_{j \neq i} V_{xi} \cdot V_{xj} + \frac{B}{2} \cdot \sum_i \sum_X \sum_{X \neq Y} V_{xi} \cdot V_{yi} + \frac{C}{2} \cdot \left(\sum_X \sum_i V_{xi} - n \right)^2 \quad (4.24)$$

$$+ \frac{D}{2} \cdot \sum_X \sum_{X \neq Y} \sum_i d_{XY} \cdot V_{XY} \cdot (V_{Y,i+1} + V_{Y,i-1})$$

Les trois premiers termes caractérisent les contraintes du problème qui interdisent aux réseaux les états illégaux :

- une ville doit être trouvée dans une seule étape
- une étape ne peut contenir qu'une ville
- il ne faut attribuer que n valeurs pour constituer un tour.

Le dernier terme est le critère à minimiser :

- le parcours doit être de longueur minimale.

Les paramètres A, B, C, D sont définis positifs et pondèrent l'importance de chaque terme vis à vis des autres. Des valeurs convenables de ces paramètres sont fournies dans l'article [Hopfield & Tank, 1985].

La phase d'identification Elle permet de déterminer les poids synaptiques T_{ij} et les valeurs des entrées externes I_i de sorte que les états stables du réseau soient les solutions recherchées au problème posé. La fonctionnelle (4.24) préalablement définie contient les informations nécessaires pour rendre états stables du réseau ses minima locaux. Il suffit alors d'identifier l'énergie du réseau (4.11) et la fonctionnelle (4.24) :

- les coefficients des termes quadratiques représentent les poids synaptiques T_{ij} qui sont symétriques.
- Les coefficients des termes linéaires représentent les entrées externes I_i .

Les expressions sont alors les suivantes :

$$T_{X_i, Y_j} = -A \cdot (1 - \delta_{i,j}) \cdot \delta_{X,Y} - B \cdot (1 - \delta_{X,Y}) \cdot \delta_{i,j} - C - D \cdot d_{X,Y} \cdot (\delta_{i+1,j} + \delta_{i-1,j}) \quad (4.25)$$

$$I_{X_i} = C \cdot n \quad (4.26)$$

où δ_{ij} est la fonction de Kronecker.

Des résultats convaincants ont été apportés dans une bonne proportion de cas [Hopfield & Tank, 1985]. Toutefois, certaines difficultés restreignent cette méthode. La détermination des paramètres A, B, C, D n'est pas simple pour assurer un résultat correcte au réseau. La solution obtenue est un minimum local de la fonction d'énergie qui dépend de l'état initial du réseau. Dans ce dernier cas, il est possible d'améliorer ce minimum, voire même atteindre le minimum global, en utilisant des techniques heuristiques telles que *le recuit simulé*.

4.3.4.2 Réseaux récurrents et recuit simulé

Ce principe du recuit simulé est a priori indépendant des réseaux de neurones. Pourtant, Hopfield [Hopfield & Tank, 1985] démontra comment le mettre en oeuvre sur son réseau. Dans le modèle discret, un neurone change d'état dès que la condition (4.12) sur la variation d'énergie est vérifiée. L'inconvénient de cette approche déterministe est que, pour une condition initiale donnée, le réseau se coince dans un minimum local. L'apport du recuit simulé peut permettre au réseau d'effectuer des changements d'état *augmentant* l'énergie. Un neurone fonctionnant suivant une règle stochastique (cf chapitre 2 §2.2.3) envisage simultanément les deux états possibles. Lorsque l'énergie augmente, le changement d'état est effectué selon une probabilité. De cette façon, on s'assure que le réseau converge vers un minimum global et non vers un des minima locaux.

Cette approche appliquée au problème précédent du voyage de commerce a montré une diminution sensible de la longueur du parcours.

Ce modèle de réseaux nécessite cependant un temps de calcul plus important que le modèle déterministe. De ce fait, selon l'objectif à atteindre, on privilégie souvent l'approche déterministe qui offre un bon compromis qualité de la solution/temps de calcul.

4.4 Conclusion

De nombreux problèmes d'optimisation ont été abordés par des réseaux récurrents, que ce soit dans le traitement d'image [Lin, Liao, Tsao, & Lingutla, 1991], les télécommunications [Page & Tagliarini, 1988], informatique [Herault, 1989] et traitement du signal, conception de circuits électroniques [Hopfield & Tank, 1986], etc

Dans le chapitre suivant, nous allons développer un modèle généraliste de réseau pour la minimisation de fonctionnelles, sur l'exemple de celles décrites au chapitre 2, en identifiant la variable solution z à une somme d'états de neurones.

Chapitre 5

Méthodologie neuronale proposée

Au cours du chapitre précédent, nous avons présenté une architecture neuronale du type réseau récurrent qui est capable de minimiser des fonctionnelles de forme quadratique sur un principe différent des approches heuristiques. Par ailleurs, comme cela a été vu au chapitre 2, de nombreux problèmes de traitement d'images ont été modélisés en termes de fonction d'énergie pour la plupart quadratique. Au cours de ce chapitre, nous développons une *méthodologie neuronale* dont le principe de base consiste à coder la solution des problèmes d'optimisation en termes d'états de neurone, qui sont les variables discrètes d'un espace euclidien.

5.1 Définition d'une image

D'une façon générale, une image de dimensions $X \times Y$ est formée en affectant à chaque point de coordonnées (x,y) du plan de l'image une valeur numérique correspondant à son intensité lumineuse $f(x,y)$, également appelée fonction de *luminance* avec :

$$0 \leq f(x,y) \leq \theta \quad \text{avec } (x,y) \in (X,Y) \quad (5.1)$$

où θ est un paramètre à valeurs réelles.

Aucun système actuellement n'est capable de traiter directement une image, à l'exception des systèmes optiques. L'image est alors discrétisée en surfaces élémentaires appelées *pixel*. Le réseau de discrétisation est de façon générale une grille rectangulaire de pixels de dimensions (a,b) à l'intérieur desquels la luminance est constante (la luminance est égale à l'intensité lumineuse par unité de surface). Une image de dimension $X \times Y$ est donc constituée de la superposition de 'L' lignes de hauteur 'b' chacune et de 'M' pixels de longueur 'a' par ligne avec les relations :

$$Y=L \cdot b \quad X= M \cdot a \quad (5.2)$$

De même, pour pouvoir traiter l'image par un ordinateur numérique, la fonction intensité est discrétisée. Cette numérisation fait intervenir deux notions :

- l'échantillonnage
- la quantification

Cette nature discrète de l'image conduit en général à l'utilisation de version discrète de fonctions fréquemment obtenues par le biais des différences finies.

5.2 Méthodologie neuronale proposé pour l'optimisation

5.2.1 Modèle de fonctionnelle à minimiser

En faisant abstraction de la nature discrète de l'image et en considérant le cas continu, les problèmes de vision se formulent de façon générale au moyen de deux contraintes sur le modèle suivant :

$$E = \iint \left\{ (A \cdot z - b)^2 + \alpha^2 \cdot g(z) \right\} \cdot dx \cdot dy \quad (5.3)$$

où le premier terme représente l'attache aux données à résoudre avec A, l'opérateur mathématique, z, la variable solution et b, les données, et le second membre g(z) traduit la contrainte de lissage qui s'exprime sous la forme d'un opérateur différentiel dont deux des choix les plus communs sont :

➤ l'opérateur différentiel du premier ordre

$$\left(\frac{\partial z}{\partial x} \right)^2 + \left(\frac{\partial z}{\partial y} \right)^2 \quad (5.4)$$

➤ l'opérateur Laplacien

$$\nabla^2 z \quad \text{où} \quad \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (5.5)$$

Si l'on retranscrit la fonctionnelle (5.3) dans un domaine discret, la solution finale doit minimiser la somme des termes d'erreur sur l'ensemble des points (i,j) de l'image :

$$E = \sum_{i=1}^L \sum_{j=1}^M \left\{ \left[a_{i,j} \cdot z_{i,j} - b_{i,j} \right]^2 + \alpha^2 \cdot g(z_{i,j}) \right\} \quad (5.6)$$

où $a_{i,j}$, $z_{i,j}$, $b_{i,j}$ traduisent respectivement la valeur du coefficient de l'opérateur A, la variable solution et la donnée au pixel de coordonnées (i,j).

Le terme de régularisation variationnelle $g(z_{i,j})$ fait intervenir des dérivées partielles du premier ou du second ordre de la variable $z_{i,j}$. Or, de par cette nature discrète, il est nécessaire d'approcher en chaque point cette variable au moyen de fonctions analytiques continues différentiables afin de pouvoir déterminer ces dérivées partielles.

5.2.2 Estimation des dérivées partielles

Soit $f(i,j)$ une fonction discrète. Afin de simplifier les notations, l'origine du système de coordonnées locales est supposé centré au point d'intérêt, c'est à dire en (i,j). Comme il est bien connu en géométrie différentielle élémentaire, une surface peut être approchée localement par une fonction d'ordre n du vecteur de coordonnées (p,q) :

$$f(p, q) = f_0 + \frac{\partial f}{\partial p} \cdot p + \frac{\partial f}{\partial q} \cdot q + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial p^2} \cdot p^2 + \frac{1}{2} \cdot \frac{\partial^2 f}{\partial q^2} \cdot q^2 + \frac{\partial^2 f}{\partial p \partial q} \cdot p \cdot q + \dots \quad (5.7)$$

En reformulant l'équation (5.7), l'approximation locale de cette fonction sur une petite région au moyen d'un nombre limité de polynômes bidimensionnels peut s'écrire de façon plus générale :

$$f(p, q) = \sum_{h=0}^H b_h \cdot \varphi_h(p, q) \quad (5.8)$$

où (H+1) représente le nombre de polynômes bidimensionnels. $\varphi_h(p, q)$ sont les fonctions de base sur le domaine de définition \mathcal{D} choisi dont un exemple est présenté par la Figure 1.

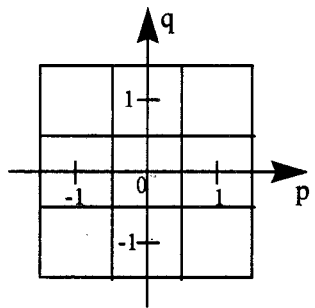


Figure 1 : Domaine de définition \mathcal{D}

Les coefficients de pondérations b_h des fonctions de base précédentes sont obtenues en minimisant l'erreur quadratique moyenne entre la fonction $f(p, q)$ et son approximation. On obtient alors :

$$b_h = \frac{\sum_p \sum_q \varphi_h(p, q) \cdot f(p, q)}{\sum_p \sum_q \varphi_h(p, q) \cdot \varphi_h(p, q)} \quad \text{pour } h = 0, \dots, H \quad (5.9)$$

Une interprétation intéressante de la relation (5.9) est de considérer les coefficients b_h , comme résultant d'une opération de convolution entre la fonction et un noyau de convolution M_h tels que :

$$b_h = M_h(p,q)**f(p,q) \quad (5.10)$$

où ** représente la convolution bidimensionnelle.

Prenons le cas concret d'un domaine de dimension 3x3, centré à l'origine comme l'illustre la Figure 1. Pour estimer les dérivées partielles du second ordre, il est nécessaire de définir 9 fonctions orthogonales. Autrement dit, H est égale à 8. Ce choix permet en effet d'utiliser des fonctions polynomiales d'ordre supérieur à 1. Les fonctions de base sont alors les suivantes :

$$\varphi_0(p,q) = 1 \quad (5.11a)$$

$$\varphi_1(p,q) = p \quad (5.11b)$$

$$\varphi_2(p,q) = q \quad (5.11c)$$

$$\varphi_3(p,q) = p^2 - \frac{2}{3} \quad (5.11d)$$

$$\varphi_4(p,q) = q^2 - \frac{2}{3} \quad (5.11e)$$

$$\varphi_5(p,q) = p \cdot q \quad (5.11f)$$

$$\varphi_6(p,q) = \left(p^2 - \frac{2}{3}\right) \cdot q \quad (5.11g)$$

$$\varphi_7(p,q) = \left(q^2 - \frac{2}{3}\right) \cdot p \quad (5.11h)$$

$$\varphi_8(p,q) = \left(p^2 - \frac{2}{3}\right) \cdot \left(q^2 - \frac{2}{3}\right) \quad (5.11i)$$

Les coefficients b_h , $h=0, \dots, 8$ s'obtiennent toujours par la relation (5.9) et selon la convention du repère de la Figure 1, les noyaux de convolution se définissent :

$$\begin{aligned}
 M_0 &= \frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} & M_1 &= \frac{1}{6} \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} & M_2 &= \frac{1}{6} \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{pmatrix} \\
 M_3 &= \frac{1}{6} \begin{pmatrix} 1 & -2 & 1 \\ 1 & -2 & 1 \\ 1 & -2 & 1 \end{pmatrix} & M_4 &= \frac{1}{6} \begin{pmatrix} 1 & 1 & 1 \\ -2 & -2 & -2 \\ 1 & 1 & 1 \end{pmatrix} & M_5 &= \frac{1}{4} \begin{pmatrix} -1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{pmatrix} & (5.12) \\
 M_6 &= \frac{1}{4} \begin{pmatrix} -1 & 0 & 1 \\ 2 & 0 & -2 \\ -1 & 0 & 1 \end{pmatrix} & M_7 &= \frac{1}{4} \begin{pmatrix} 1 & -2 & 1 \\ 0 & 0 & 0 \\ -1 & 2 & -1 \end{pmatrix} & M_8 &= \frac{1}{4} \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}
 \end{aligned}$$

Si l'on identifie la relation (5.9) avec (5.10) pour $h=8$, on obtient alors au point $(0,0)$ du domaine de définition les dérivées partielles du premier et second ordre :

$$\frac{\partial f}{\partial p} = b_1 - \frac{2}{3} \cdot b_6 = \left(M_1 - \frac{2}{3} \cdot M_6 \right) ** f(p,q) \quad (5.13a)$$

$$\frac{\partial f}{\partial q} = b_2 - \frac{2}{3} \cdot b_7 = \left(M_2 - \frac{2}{3} \cdot M_7 \right) ** f(p,q) \quad (5.13b)$$

$$\frac{\partial^2 f}{\partial p^2} = 2 \cdot \left(b_3 - \frac{2}{3} \cdot b_8 \right) = 2 \cdot \left(M_3 - \frac{2}{3} \cdot M_8 \right) ** f(p,q) \quad (5.13c)$$

$$\frac{\partial^2 f}{\partial q^2} = 2 \cdot \left(b_4 - \frac{2}{3} \cdot b_8 \right) = 2 \cdot \left(M_4 - \frac{2}{3} \cdot M_8 \right) ** f(p,q) \quad (5.13d)$$

$$\frac{\partial^2 f}{\partial p \partial q} = b_5 = M_5 ** f(p,q) \quad (5.13e)$$

L'approximation du Laplacien de la relation (5.5) s'exprime alors :

$$\nabla^2 f(p,q) = 2 \cdot \left(b_3 + b_4 - \frac{4}{3} \cdot b_8 \right) = M_{\nabla} ** f(p,q) \quad (5.14a)$$

$$\text{où} \quad M_{\nabla} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (5.14b)$$

Dans le cas où l'on souhaiterait réduire l'influence du bruit sur les dérivées partielles, il suffit d'augmenter la taille du domaine de définition (exemple 5x5). Il est intéressant de noter que ces dérivées partielles résultent d'une opération de convolution entre la fonction f et des noyaux particuliers de dimension relative au domaine de définition choisi. Ces estimations des dérivées partielles peuvent être interprétées comme des opérations de filtrage linéaire. Pour la suite de notre étude, nous reformulons l'expression matricielle des dérivées partielles sous l'aspect suivant :

$$\sum_p \sum_q d_{p,q} \cdot f(p,q) \quad (5.15)$$

où $d_{p,q}$ représente les différents coefficients de la matrice relative à la dérivée approchée et (p,q) , les valeurs des coordonnées dans le domaine de définition \mathcal{D} .

Revenons à notre problème d'optimisation et, à titre d'exemple, prenons l'option de l'opérateur différentielle du premier ordre (5.4) comme contrainte de lissage. En reportant cette contrainte $g(z)$ et en substituant les dérivées partielles par leur approximation (5.15) dans la fonctionnelle (5.6), on obtient la formulation générale de notre problème :

$$E = \sum_i \sum_j \left\{ \left[a_{i,j} \cdot z_{i,j} - b_{i,j} \right]^2 + \alpha^2 \cdot \left[\left(\sum_p \sum_q d_{p,q}^x \cdot z_{i+p,j+q} \right)^2 + \left(\sum_p \sum_q d_{p,q}^y \cdot z_{i+p,j+q} \right)^2 \right] \right\} \quad (5.16)$$

où $d_{p,q}^x$ et $d_{p,q}^y$ représentent respectivement les coefficients des dérivées partielles en x et en y .

5.2.3 Représentation de la variable

Nous proposons un réseau de type Hopfield discret contenant N neurones redondants pour représenter la variable $z_{i,j}$ de la fonctionnelle à minimiser comme l'illustre la Figure 2. Par la suite, nous emploierons le terme de *sous-réseau* pour exprimer un réseau évoquant la variable z en un point de coordonnées (i,j) . Le modèle neuronal global se constitue alors de $L \times M$ sous-réseaux, autrement dit $L \times M \times N$ neurones mutuellement interconnectés où $L \times M$ représente les dimensions de l'image. Soit $V = \{V_{i,j,h}, \text{ où } 1 \leq i \leq L, 1 \leq j \leq M, \text{ et } 1 \leq h \leq N\}$ une série d'états binaires du sous-réseau de neurones avec $V_{i,j,h}$

représentant l'état du neurone h à la position (i,j) . La variable $z_{i,j}$ est déduite en utilisant une simple somme [Takeda & Goodman, 1986] des variables d'états des neurones du sous-réseau :

$$z_{i,j} = k \cdot \sum_{h=1}^N V_{i,j,h} \quad (5.17)$$

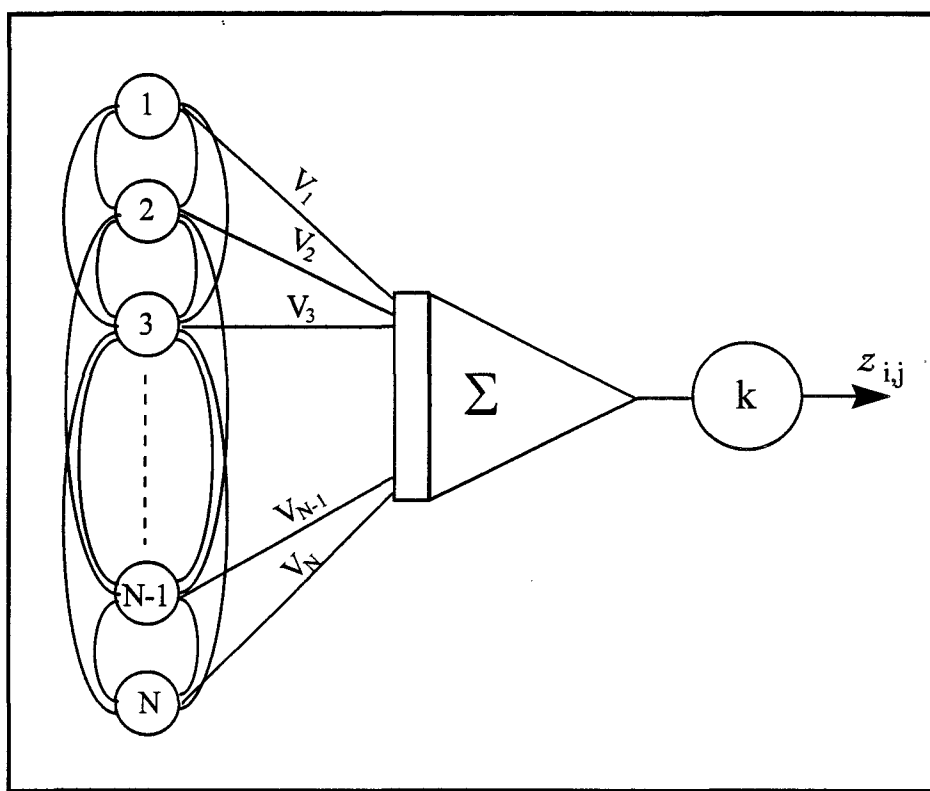


Figure 2 : Représentation schématique de la variable z en un point (i,j)

Le coefficient k est un paramètre de quantification qui transforme l'activation d'une unité supplémentaire dans l'espace des variables d'états en une variation de la valeur z dans l'espace de ses solutions. Le produit $k \cdot N$ définit alors la *dynamique* de la variable z .

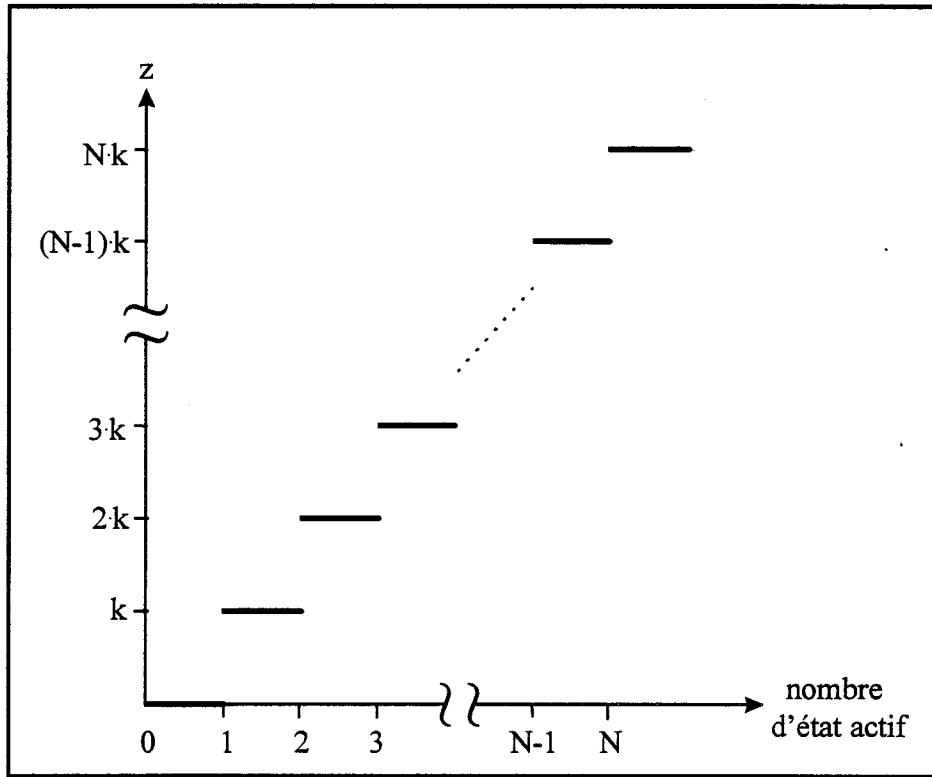


Figure 3 : Représentation quantifiée de la solution z

Il est intéressant de noter aussi que ce système de représentation de la variable z dans un espace d'états de neurones est redondant. Considérons l'exemple d'un sous-réseau de 10 neurones au sein desquels 3 d'entre-eux sont actifs, déterminant ainsi la valeur de z .

Il apparaît évident de dénombrer $\frac{10!}{3!(10-3)!} = 120$ configurations stables (ou états à

énergie minimale) possibles dans ce sous-réseau. De façon plus générale pour un sous-réseau de N neurones, si $X_{i,j}$ ($X_{i,j} \leq N$) neurones sont nécessaires pour représenter la valeur de $z_{i,j}$, il existe

$$\frac{N!}{X_{i,j}!(N - X_{i,j})!} \tag{5.18}$$

états stables. Soit, pour une image de dimension $L \times M$:

$$\prod_{i=1}^L \prod_{j=1}^M \frac{N!}{X_{i,j}!(N - X_{i,j})!} \tag{5.19}$$

La redondance de ces états stables procure au réseau davantage de chance d'atteindre une solution correct plus rapidement, et procure une plus grande tolérance à l'erreur [Takeda & Goodman, 1986]. Un autre intérêt, sous un aspect plus technologique (conception d'un circuit neuronique), de cette représentation est que la défaillance ou la perte d'un neurone n'empêche pas le réseau de fonctionner mais tout au plus restreint la dynamique de la variable z .

5.2.4 Estimation des paramètres du modèle

Seules les formulations principales sont présentées ici. Les détails calculatoires sont décrits dans l'Annexe II. Les paramètres du modèle neuronal proposé, les poids synaptiques et entrées externes, se déterminent en termes de fonction d'énergie du réseau. Sur le modèle de J. Hopfield et D. W. Tank [Hopfield & Tank, 1985], la fonction d'énergie du réseau relatif au problème (5.6) peut s'écrire :

$$E = -\frac{1}{2} \sum_{i=1}^L \sum_{j=1}^M \sum_{m=1}^L \sum_{n=1}^M \sum_{h=1}^N \sum_{l=1}^N T_{i,j,h;m,n,l} \cdot V_{m,n,l} \cdot V_{i,j,h} - \sum_{i=1}^L \sum_{j=1}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (5.20)$$

En remplaçant z_{ij} par son approximation (5.15) dans la fonctionnelle (5.16) à minimiser, nous obtenons :

$$E = \sum_{i=1}^L \sum_{j=1}^M \left\{ \left[a_{i,j} \cdot \left(k \cdot \sum_{h=1}^N V_{i,j,h} \right) - y_{i,j} \right]^2 + \alpha^2 \cdot \left[\left(\sum_p \sum_q d_{p,q}^x \cdot \left(k \cdot \sum_{h=1}^N V_{i+p,j+q,h} \right) \right)^2 + \left(\sum_p \sum_q d_{p,q}^y \cdot \left(k \cdot \sum_{h=1}^N V_{i+p,j+q,h} \right) \right)^2 \right] \right\} \quad (5.21)$$

En identifiant cette relation avec l'énergie du réseau (5.20), on détermine les poids des interconnexions (5.22a) et les entrées externes (5.22b) :

$$T_{i,j,h;m,n,l} = -2 \cdot k^2 \cdot \left\{ a_{i,j}^2 \cdot \delta_{(i,j),(m,n)} + \alpha^2 \cdot \left(\sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x + d_{p,q}^y \cdot d_{r,s}^y \right) \cdot \delta_{(i,j),(m,n) \oplus s} \right\} \quad (5.22a)$$

$$I_{i,j,h} = -2 \cdot k \cdot a_{i,j} \cdot b_{i,j} \quad (5.22b)$$

où $S=\{(-p+r,-q+s)\}$ représente une série d'index, $\delta_{(i,j),(m,n)}$ la fonction de Kronecker définie par :

$$\delta_{(i,j),(m,n)} = \begin{cases} 1 & \text{si } (i,j) = (m,n) \\ 0 & \text{sinon} \end{cases} \quad (5.23)$$

et

$$f_{(i,j)\oplus s} = \begin{cases} f_{i-p+r,j-q+s} & \text{si } p,q,r,s \in \mathcal{D} \\ 0 & \text{sinon.} \end{cases} \quad (5.24)$$

De par la définition sur un domaine de dimensions 3×3 choisi pour approcher les dérivées partielles, la série d'index S va s'étendre entre les couples $(-2, -2)$ et $(2, 2)$ pour générer un pavé de dimensions 5×5 centré en (i, j) , Figure 4.

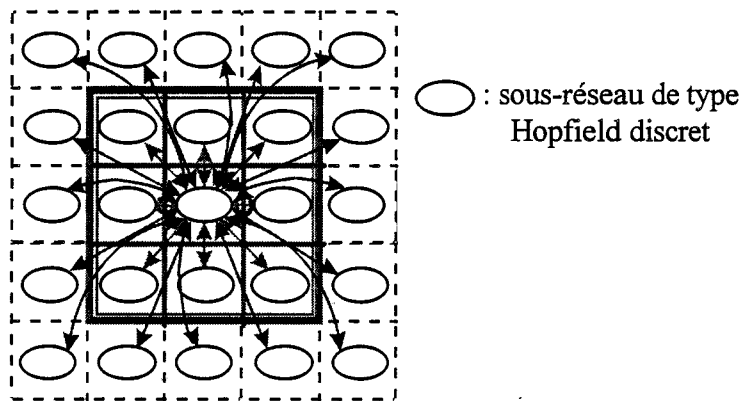


Figure 4 : Connexions entre sous-réseaux dues au terme variationnel

A partir de la relation (5.22b), nous pouvons constater que les entrées externes dépendent des données b_{ij} du problème. Ceci confirme bien que les informations extérieures (fournies par le problème), qui vont permettre au réseau de converger vers la solution recherchée, sont transmises par le biais de ces entrées. Il est intéressant de noter qu'au sein d'un même sous-réseau localisé en un point (i,j) , la valeur de l'entrée externe de chacun des neurones le constituant est identique. Du point de vue des interconnexions, il est possible de les distinguer en deux grandes catégories :

- celles reliant les neurones d'un même sous-réseau, que l'on peut renoter $T_{i,j,h;i,j,k}$
- celles reliant les neurones des différents sous-réseaux, $T_{i,j,h;m,n,k}$ avec $(i,j) \neq (m,n)$

Les connexions *intra sous-réseaux* $T_{i,j,h;i,j,k}$ à valeurs négatives et identiques, symbolisent le phénomène de sommation recherché pour caractériser la variable. Toutefois, l'existence d'une connexion à valeurs négatives du neurone sur lui-même, communément appelée *auto-connexion*, a pour conséquence la non-convergence absolue du réseau.

La seconde catégorie de connexions est due aux opérateurs différentiels, Figure 4. Ils créent des liaisons qui vont répondre au critère de la contrainte de régularisation souhaitée. Ce sont ces connexions qui génèrent un réseau global de type Hopfield. En effet, en l'absence de ces poids (par exemple si $\alpha=0$), nous aurions $L \times M$ sous-réseaux indépendants. Chaque sous-réseau peut donc rechercher son minimum local de manière individuelle. Cette configuration s'apparente à la résolution locale d'un problème d'optimisation.

5.2.5 Dynamique du réseau

Une fois les paramètres $T_{i,j,h;m,n,k}$ et $I_{i,j,h}$ définis, chaque neurone (i,j,h) peut aléatoirement et de façon asynchrone évaluer son état :

$$u_{i,j,h} = \sum_m^L \sum_n^M \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} + I_{i,j,h} \quad (5.25a)$$

$$V_{i,j,h} = H(u_{i,j,h}) \quad \text{où } H \text{ est la fonction de Heaviside} \quad (5.25b)$$

Lorsqu'un état d'énergie quasi-minimum (minimum local) est atteint, la solution z est reconstruite en utilisant la relation (5.17).

Cependant, ce modèle neuronal possède des auto-connexions $T_{i,j,h;i,j,h}$ à valeurs négatives. Comme conséquence, la fonction d'énergie ne décroît pas toujours de façon monotone avec un changement d'état et la convergence du réseau n'est pas garantie [LaSalle, 1986]. En effet, définissons $\Delta V_{i,j,h}$ la variation du vecteur d'état due à la transition du neurone (i,j,h) et ΔE , la variation d'énergie qui s'ensuit par :

$$\Delta V_{i,j,h} = V_{i,j,h}^{\text{new}} - V_{i,j,h}^{\text{old}} \quad \text{et} \quad \Delta E = E^{\text{new}} - E^{\text{old}} \quad (5.26)$$

En considérant la fonction d'énergie du réseau (5.20), la variation d'énergie due au changement d'état peut s'écrire :

$$\Delta E = -\left(u_{i,j,h} \cdot \Delta V_{i,j,h} + \frac{1}{2} \cdot T_{i,j,h;i,j,h} \cdot (\Delta V_{i,j,h})^2\right) \quad (5.27)$$

Il est facile de constater que cette relation n'est pas toujours négative. Par exemple, si

$$V_{i,j,h}^{\text{old}} = 0 \quad \text{et} \quad u_{i,j,h}^{\text{old}} = \sum_m^L \sum_n^M \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} + I_{i,j,h} > 0 \quad (5.28)$$

après la fonction de seuillage (5.25b), $V_{i,j,h}^{\text{new}} = 1$ d'où $\Delta V_{i,j,h} > 0$. Donc le premier terme de (5.27) est positif. Mais $T_{i,j,h;i,j,h} < 0$ a pour conséquence, dans le cas où

$$\frac{1}{2} \cdot |T_{i,j,h;i,j,h}| > u_{i,j,h} \quad (5.29)$$

que la variation d'énergie ΔE est positive, ce qui signifie que E n'est pas une fonction de Lyapunov [Barnett & Storey, 1970].

Ainsi, il devient nécessaire d'avoir recours à des techniques qui assurent la convergence du réseau. Selon que l'on souhaite atteindre un minimum local ou global, deux règles de décision ont été développées :

- la règle déterministe
- la règle stochastique

La règle déterministe consiste à valider le nouvel état $V_{i,j,h}^{\text{new}}$ du neurone (i,j,h) si la variation d'énergie ΔE , qui fait suite à ce changement, est négative. Dans le cas contraire, $\Delta E > 0$, l'état du neurone demeure inchangé. L'autre approche, la règle stochastique, fait appel aux techniques du recuit simulé [Kirkpatrick, Gelatt & Vecchi, 1983].

Les calculs détaillés de la variation d'énergie ΔE (5.27) sont disponibles dans l'Annexe III.

La méthode proposée recouvre des travaux antérieurs. Y.T. Zhou et R. Chellappa [Zhou & Chellappa, 1988] ont en effet employé ce type de méthodologie neuronale au problème de la restauration d'images.

5.3 Exemple de la restauration

Le problème de la restauration des images posé sous la forme d'un problème d'optimisation sous contrainte a pour formulation [Andrews & Hunt, 1977] :

$$E = \frac{1}{2} \cdot \|H \cdot \tilde{F} - G\|^2 + \frac{1}{2} \cdot \lambda \cdot \|D \cdot \tilde{F}\|^2 \quad (5.30)$$

où \tilde{F} (variable solution que l'on peut associer à z de la relation 5.3) et G (les paramètres mesurés comparables à b au sein de cette même relation) dénotent respectivement les images restaurée et observée (dégradée), Figure 5. H (équivalent au terme A) traduit les *dégradations déterministes*, telles que le flou dû aux aberrations du système optique, diffraction, mouvement, ..., et D l'opérateur de régularisation. Le second terme $\|D \cdot \tilde{F}\|^2$ correspond donc à la contrainte de régularisation $g(z)$ de l'équation 5.3. Des *dégradations statistiques*, telles que le bruit dû aux capteurs d'images électroniques, ..., peuvent aussi être prises en considération. Ces phénomènes se caractérisent par l'ajout d'un bruit additif B .

Le premier terme de (5.30), ou contrainte du problème, recherche une estimation \tilde{F} de l'image F telle que $H \cdot \tilde{F}$ approche G au sens des moindres carrés [Wintz, 1977]. Tandis que le second terme est une contrainte de lissage sur la solution \tilde{F} . La constante λ détermine l'importance relative de ces deux termes.

La restauration d'une image de haute qualité à partir d'une acquisition dégradée est un problème important dans les tâches de vision primaire. Comme il l'a été précisé au début de ce chapitre, une image peut être caractérisée par une fonction à deux dimensions $f(x,y)$. Puisque l'intensité lumineuse est une quantité réelle positive et que sa valeur maximale est limitée par les caractéristiques du capteur de lumière (ou caméra), $f(x,y)$ est une fonction réelle positive finie comme définie relation (5.1) où A représente la valeur maximale de l'intensité lumineuse. La sortie du convertisseur A/N, Figure 5, procure un signal quantifié $g(i,j)$ de l'intensité lumineuse. Les valeurs de l'intensité quantifiée, ou *niveaux de gris*, s'étendent alors de 0 à Q , où Q est un entier ($Q=255$, par exemple).

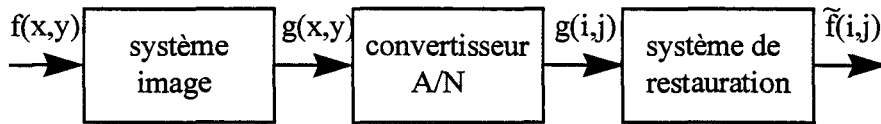


Figure 5 : système de restauration d'images digitalisées

H s'exprime en général sous forme d'un filtre de distorsion passe-bas dont une expression possible sur un domaine de définition de dimensions 3x3 est :

$$\frac{1}{2} \cdot \begin{bmatrix} 1/8 & 1/8 & 1/8 \\ 1/8 & 1 & 1/8 \\ 1/8 & 1/8 & 1/8 \end{bmatrix} \quad (5.31)$$

Si H est défini comme ci-dessus, alors le choix de l'opérateur de régularisation D est souvent un filtre passe-haut dont l'opérateur Laplacien fait partie. L'approximation de ce dernier sur un même domaine de définition que H est :

$$\frac{1}{6} \cdot \begin{bmatrix} 1 & 4 & 1 \\ 4 & -20 & 4 \\ 1 & 4 & 1 \end{bmatrix} \quad (5.32)$$

Si l'on retranscrit ces deux expressions sur le modèle de la relation (5.15), on obtient respectivement :

$$H = \sum_a \sum_b h(a, b) \quad \text{où } h(a, b) = \begin{cases} \frac{1}{2} & \text{si } a = b = 0 \\ \frac{1}{16} & \text{si } |a| \leq 1, |b| \leq 1 \text{ et } (a, b) \neq (0, 0) \end{cases} \quad (5.33)$$

$$D = \sum_p \sum_q d_{p,q}^\nabla \quad \text{où } d_{p,q}^\nabla = \begin{cases} -\frac{10}{3} & \text{si } (p, q) = (0, 0) \\ \frac{2}{3} & \text{si } (p, q) \in \{(-1, 0), (0, -1), (1, 0), (0, 1)\} \\ \frac{1}{6} & \text{si } (p, q) \in \{(-1, -1), (-1, 1), (1, -1), (1, 1)\} \end{cases} \quad (5.34)$$

La version discrétisée du problème de restauration d'images, relation (5.30), s'exprime alors par :

$$E = \frac{1}{2} \cdot \sum_{i=1}^L \sum_{j=1}^M \left(\sum_a \sum_b h_{a,b} \cdot \tilde{f}_{i+a,j+b} - g(i,j) \right)^2 + \frac{1}{2} \cdot \lambda \cdot \sum_{i=1}^L \sum_{j=1}^M \left(\sum_p \sum_q d_{p,q}^{\nabla} \cdot \tilde{f}_{i+p,j+q} \right)^2 \quad (5.35)$$

Le modèle neuronal développé pour ce problème se constitue de $L \times M \times N$ neurones où N représente le nombre de neurones par sous-réseaux. La valeur de N est alors identique à Q pour pouvoir caractériser les valeurs de niveaux de gris $\tilde{f}_{i,j}$ de l'image restaurée. La valeur du paramètre de quantification k est donc dans ce cas égale à 1. Autrement dit, à chaque état actif d'un neurone est associée une unité supplémentaire de niveau de gris. Si l'on identifie à présent la fonctionnelle (5.35) avec la fonction d'énergie du réseau (5.20), on obtient les valeurs suivantes pour les paramètres du réseau, poids synaptiques et entrées externes :

$$T_{i,j,k;m,n,l} = - \left\{ \sum_p \sum_q \sum_r \sum_s (h_{p,q} \cdot h_{r,s} + \lambda \cdot d_{p,q}^{\nabla} \cdot d_{r,s}^{\nabla}) \cdot \delta_{(i,j),(m,n) \oplus s} \right\} \quad (5.36a)$$

$$I_{i,j,k} = -g_{i,j} \cdot \sum_p \sum_q h_{p,q} \quad (5.36b)$$

Le regroupement sous une seule et même sommation des coefficients de l'approximation de la fonction de flou 'h' et du laplacien 'd[∇]' est possible puisqu'ils ont un domaine de définition identique en dimension. A titre indicatif, si l'on intègre le paramètre de bruit B dû aux dégradations statistiques, seule l'expression (5.36b) des entrées externes serait modifiée et deviendrait :

$$I_{i,j,k} = -(g_{i,j} - b_{i,j}) \cdot \sum_p \sum_q h_{p,q} \quad (5.37)$$

Des exemples de résultats obtenus par cette approche sont donnés dans l'article de [Zhou & Chellappa, 1988].

5.4 Conclusion

Ce chapitre a introduit une nouvelle approche pour la minimisation de fonctionnelle de type quadratique sur le modèle de la relation (5.3). L'idée essentielle est de représenter la solution d'un problème d'optimisation dans un espace d'état de neurones. La solution se code alors sous forme d'une somme de ces états binaires. L'évolution temporelle du réseau récursif constitué de ces neurones aboutit à une solution locale à énergie minimale.

La solution que nous proposons, reprend et généralise une première approche de ce type décrite par Y.T. Zhou et R. Chellappa sur le problème particulier de la restauration d'images.

Nous développerons par la suite une analyse plus approfondie de la méthode proposée sur le cas bien plus complet de l'estimation du mouvement apparent. En effet, cette classe de problèmes correspond à l'estimation d'un vecteur, constitué de deux composantes à valeurs réelles et dimensionnées.

Chapitre 6

Estimation du mouvement

Lors du chapitre précédent, nous avons proposé un réseau générique pour la minimisation de fonctionnelle de type quadratique. L'exemple particulier (variable discrète, positive et sans dimension) de la restauration d'images a été présenté. Nous allons à présent appliquer cette méthodologie neuronale au problème de l'estimation du mouvement apparent, problème plus vaste, puisque la solution recherchée est une vitesse, variable vectorielle, continue à valeurs positives ou négatives, et dimensionnée.

Parmi les nombreuses méthodes qui permettent une estimation du mouvement apparent dans des séquences d'images, l'approche différentielle conduit à une formulation qui correspond à une minimisation de fonctionnelle du type de celle prise en compte par la structure du réseau neuronal proposé.

La Figure 1 présente un essai de description schématique des différentes approches existantes en analyse du mouvement dans les séquences d'images. La position occupée dans ce schéma par les méthodes différentielles montre qu'il s'agit de méthodes de niveaux intermédiaires entre les approches bas niveaux et haut niveaux.

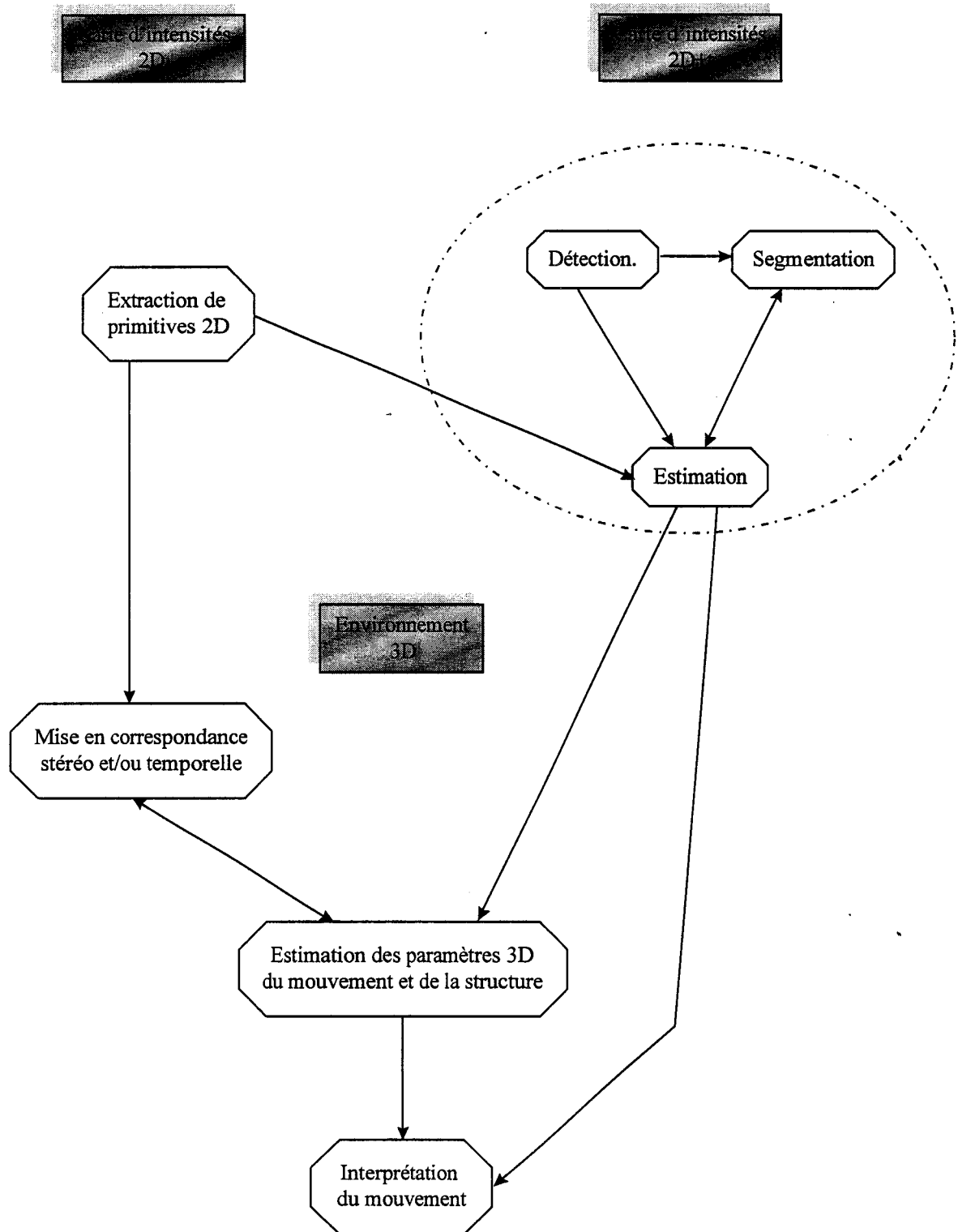


Figure 1 : Divisions majeures de la recherche en analyse du mouvement

6.1 Mesure du mouvement apparent

On définit par mouvement apparent (ou champ de vitesses apparent) le mouvement 2D perçu dans l'image au travers des variations temporelles de l'intensité lumineuse, dues aux mouvements 3D relatifs à la scène. Ce champ ne caractérise pas le véritable mouvement 2D qui serait la projection sur le plan image du mouvement 3D que l'on dénommera mouvement projeté. Ce dernier est en effet inaccessible par la mesure. Toutefois, même si ces deux champs sont différents, ils demeurent néanmoins très liés [Verri & Poggio, 1986]. Ce champ apparent peut être appréhendé par d'autres techniques que celles des méthodes différentielles. C'est en particulier le cas avec les *méthodes de mise en correspondance* et les *méthodes par transformée*.

Le groupe des méthodes de mise en correspondance comprend des techniques basées sur une fonction de corrélation ou de similarité entre blocs [Aggarwal, Davis, & Martin, 1981] [Burt, Yen & Xu, 1982] [Shirai, 1987] qui sont très limitées (hypothèse de mouvements apparents de type translationnel), ou des techniques qui, après extraction de primitives (segments de droite, contours, coins, régions,...) dans chaque image, établissent la carte du champ "éparse" par des procédures de relaxation ou de mise en correspondances de graphes [Barnard & Thompson, 1980] [Cheng & Huang, 1984] [Shirai, 1987].

Les méthodes par transformées sont d'un développement plus récent. Ces dernières opèrent dans un espace autre que l'espace spatio-temporel de l'image comme celui du fréquentiel. L'apport d'informations complémentaires peut ainsi permettre d'obtenir des estimations de qualité. Néanmoins, ces méthodes se distinguent en général par de lourds calculs, ce qui réduit considérablement le champ d'application potentiel. De telles approches sont présentées dans différents articles [Samy & Bozzo, 1984] (Transformée de Hough), [Fleet & Jepson, 1990] (Transformée de Fourier), [Daugmann, 1988] (Transformée en ondelettes), [Heeger, 1988] (filtre de Gabor), [Jacobson & Wechsler, 1988] (filtre de Wigner-Ville).

Nous allons à présent exposer de façon plus détaillée les méthodes différentielles, en particulier *l'équation de contrainte du mouvement apparent, E.C.M.A.*, pour aboutir à la notion de *flot optique* (*Optical Flow* en anglais).

6.1.1 les méthodes différentielles

Si on considère un domaine D connexe strictement intérieur à un système S animé d'un mouvement de vitesse \vec{w} , il est possible d'associer à cet ensemble une loi générale dite de conservation des milieux continus qui s'exprime sous la forme :

$$\frac{d}{dt} \int_D \mathcal{R}_i \cdot dV + \int_{\partial D} r_i \cdot d\sigma = \int_D R_i \cdot dV \quad (6.1)$$

où \mathcal{R}_i , R_i et r_i représentent trois grandeurs (scalaires, vectorielles ou tensorielles du second ordre) associées dans l'énoncé de la loi. $\frac{d}{dt}$ désigne la dérivée particulière, c'est à dire une dérivée par rapport au temps d'une grandeur définie sur un ensemble de particules que l'on suit dans son mouvement. Cette dérivée s'identifie à une dérivée partielle par rapport au temps lorsqu'on travaille en variables de Lagrange (ensemble de coordonnées associées à une particule) et s'exprime par la relation suivante en variables d'Euler :

$$\frac{d}{dt} \cdot f = \frac{\partial f}{\partial t} + \vec{w} \cdot \text{grad}(f) \quad (6.2)$$

Si les grandeurs \mathcal{R}_i , R_i et r_i sont continûment dérivables sur S , on peut associer à la loi de conservation (6.1) un système d'équations aux dérivées partielles :

$$\frac{\partial \mathcal{R}_i}{\partial t} + \text{div}(\mathcal{R}_i \cdot r_{ij}) = R_i \quad (6.3)$$

L'application de cette loi de conservation à la masse fournit une équation aux dérivées partielles :

$$\frac{\partial \rho}{\partial t} + \text{div}(\rho \cdot \vec{w}) = 0 \quad (6.4)$$

connue sous le nom d'équation de continuité. Si on développe le second membre, on aboutit à la nouvelle formulation :

$$\frac{\partial \rho}{\partial t} + \left\{ \rho \cdot \text{div}(\vec{w}) + \vec{w} \cdot \text{grad}(\rho) \right\} = 0 \quad (6.5)$$

Dans le cas particulier d'un milieu incompressible, J. Fitzpatrick [Fitzpatrick, 1988] montra à travers ces travaux, repris ensuite par S. Song et R. Leahy [Song & Leahy, 1991], que la relation (6.5) s'identifie à la classique équation de contrainte du flot optique [Fennema & Thompson, 1979] [Horn & Schunck, 1981] sur des séquences d'images dont l'intensité des pixels est proportionnelle à la densité ρ d'une grandeur conservée lors du déplacement. Dans ces conditions, la divergence des vitesses est en effet nulle ($\text{div}(\vec{w}) = 0$) et l'équation (6.5) devient :

$$\frac{\partial \rho}{\partial t} + \vec{w} \cdot \text{grad}(\rho) = 0 \quad (6.6)$$

Les images de densité d'un milieu continu forment un groupe d'images très intéressant pour l'analyse du mouvement. D'abord d'un point de vue applicatif, puisqu'elles sont utilisées en médecine (tomographie rayons X, ...) mais aussi d'un point de vue théorique puisque depuis les travaux de J. Fitzpatrick [Fitzpatrick, 1985] [Fitzpatrick, 1988], on connaît l'existence d'une transformation géométrique dans les images de densité correspondant au déplacement des objets.

Dans le cas où ce sont des images de type irradiance, on assimile l'intensité du pixel f à la grandeur ρ . Dans ce contexte, il n'existe plus nécessairement de transformation géométrique correspondant au déplacement de l'objet dans l'image. On fait cependant l'hypothèse que les variations de la fonction intensité f ne sont dues qu'au mouvement au sein de l'image, ce qui implique la validation de l'hypothèse de conservation. Si on substitue ρ par f dans (6.6), on obtient [Fennema & Thompson, 1979] [Horn & Schunck, 1981] :

$$\vec{\nabla} f \cdot \vec{w} = -f_t \quad (6.7)$$

où $\vec{\nabla} f = (f_x, f_y)^T$ représente le gradient spatial de la fonction d'intensité avec

$f_x = \frac{\partial f}{\partial x}$, $f_y = \frac{\partial f}{\partial y}$ et $f_t = \frac{\partial f}{\partial t}$, la dérivée temporelle de cette même fonction. Cette

équation est connue sous le nom d'*Equation de Contrainte du Mouvement Apparent*, que l'on notera par la suite *E.C.M.A.*. L'ensemble des vecteurs vitesse \vec{w} obtenus est appelé *flot optique* (tiré de l'expression anglaise *Optical flow*).

Les problèmes posés par l'E.C.M.A. peuvent se résumer en quatre points essentiels [Schunck, 1986] [Verri & Poggio, 1986] [Thompson & Kearny, 1987] [Bouthemy, 1989] :

➤ Phénomènes d'ombrage

➤ Propriétés de la fonction intensité :

- ◆ fortes textures : non respect de l'hypothèse de linéarité locale de la fonction intensité.
- ◆ zones uniformes : gradients spatiaux nuls, pas suffisamment d'informations.

➤ Comportement aux frontières :

- ◆ contours d'intensité : dans les zones de contraste fort et abrupt de l'intensité lumineuse, cette intensité ne présente pas un profil linéaire, même localement.
- ◆ contours d'occlusion : à position spatiale fixe, on change d'objet d'un instant au suivant; la mesure du gradient temporel de l'intensité lumineuse ne peut plus être analytiquement reliée aux variations de l'intensité.

➤ Restrictions sur le mouvement :

- ◆ grands déplacements : problème des occlusions; non recouvrement des objets d'un instant à l'autre.

Différentes approches ont été proposées pour pallier nombre de ces problèmes présentés par l'E.C.M.A. dans la mesure du mouvement apparent. Ainsi, l'emploi de techniques de multirésolution [Enkelmann, 1986] permet d'accéder à une mesure du mouvement fiable pour des déplacements de grande amplitude. La modélisation markovienne [Heitz & Bouthemy, 1990] [Konrad & Dubois, 1990], pour sa part, intègre plusieurs sources de mesure du mouvement, et spécifie des interactions spatiales locales complexes (faisant intervenir par exemple vecteurs vitesse et discontinuités de mouvement).

Il est bien connu que le problème (6.7) est mal posé au sens rappelé chapitre 2, puisque l'équation de l'E.C.M.A. ne permet d'obtenir une solution injective (critère 2, §2.2.2), et que compte tenu de l'imprécision de l'estimation des gradients, la solution n'est pas stable (critère 3, §2.2.2).

Donc on a classiquement recours à une technique de régularisation qui amène à ajouter une contrainte supplémentaire. Cette contrainte présente le double avantage d'induire un lissage du champ estimé, et de préserver une estimation cohérente dans les zones où les observations locales de mouvement ne sont plus significatives. Dans cet optique, deux techniques fondamentales ont été abordées pour mener l'estimation :

- le premier groupe concerne les approches se basant sur des méthodes variationnelles [Horn & Schunck, 1981] [Hildreth, 1984b] [Nagel, 1987]. Concrètement, ces approches conduisent à minimiser conjointement l'E.C.M.A., relatif à l'adéquation du champ aux variations spatio-temporelles de l'intensité, et une contrainte de lissage, favorisant la continuité du champ de vecteurs.
- l'autre technique consiste à aborder le problème de l'estimation suivant une approche stochastique. L'introduction de contraintes se fait au travers de modèles statistiques appropriés (estimation d'un processus stochastique 2D [Rougée, 1987], modélisation markovienne avec prise en compte des discontinuités de mouvement [Heitz & Bouthemy, 1993]).

La résolution du problème est alors en général obtenue à partir de la recherche d'une solution approchée sous forme d'une minimisation non contrainte d'une fonctionnelle dont l'exemple le plus classique est la fonctionnelle de Horn & Schunk [Horn & Schunck, 1981] dont l'expression est la suivante :

$$E = \iint \left\{ (f_x \cdot w_x + f_y \cdot w_y + f_t)^2 + \alpha^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_x}{\partial y} \right)^2 + \left(\frac{\partial w_y}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 \right] \right\} \cdot dx \cdot dy \quad (6.8)$$

α est le facteur de régularisation qui exprime le poids relatif attribué au terme de lissage vis à vis du terme de contrainte aux données, l'E.C.M.A..

6.1.2 Le flot optique par Horn & Schunck

Dans leur article, les deux auteurs ont recours au calcul variationnel (cf Annexe IV) pour aboutir au système d'équations partielles suivant :

$$\begin{aligned}\alpha^2 \cdot \nabla^2 w_x &= f_x \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t) \\ \alpha^2 \cdot \nabla^2 w_y &= f_y \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t)\end{aligned}\tag{6.9}$$

6.1.2.1 Estimation des dérivées partielles de la fonction intensité

Il est important que les estimations de f_x , f_y et f_t soient consistantes pour obtenir un champ de vitesses correct. Pour cela, l'approximation des dérivées partielles se fera sur un cube de dimensions $2 \times 2 \times 2$, Figure 2. Concrètement, les gradients spatiaux sont obtenus en moyennant les deux estimations faites dans un pavé de dimensions 2×2 sur l'ensemble des deux images. Ce principe peut s'étendre sur un horizon temporel plus vaste. Ceci aura pour effet de diminuer l'effet du bruit pour l'estimation du champ de vitesse. L'expression des dérivées spatio-temporelles est alors la suivante :

$$\begin{aligned}f_x &\approx \frac{1}{2} \cdot (f_x(t) + f_x(t+1)) \\ &\approx \frac{1}{4} \cdot \left\{ (f_{i,j+1,t} - f_{i,j,t} + f_{i+1,j+1,t} - f_{i+1,j,t}) + (f_{i,j+1,t+1} - f_{i,j,t+1} + f_{i+1,j+1,t+1} - f_{i+1,j,t+1}) \right\}\end{aligned}\tag{6.10a}$$

$$\begin{aligned}f_y &\approx \frac{1}{2} \cdot (f_y(t) + f_y(t+1)) \\ &\approx \frac{1}{4} \cdot \left\{ (f_{i+1,j,t} - f_{i,j,t} + f_{i+1,j+1,t} - f_{i,j+1,t}) + (f_{i+1,j,t+1} - f_{i,j,t+1} + f_{i+1,j+1,t+1} - f_{i,j+1,t+1}) \right\}\end{aligned}\tag{6.10b}$$

$$f_t \approx \frac{1}{4} \cdot \left\{ f_{i,j,t+1} - f_{i,j,t} + f_{i,j+1,t+1} - f_{i,j+1,t} + f_{i+1,j,t+1} - f_{i+1,j,t} + f_{i+1,j+1,t+1} - f_{i+1,j+1,t} \right\}\tag{6.10c}$$

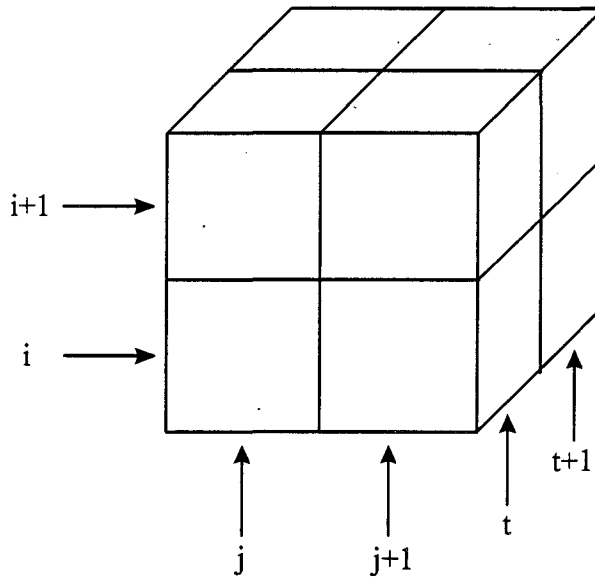


Figure 2 : Estimation des trois dérivées partielles en un point (i, j, t) .
L'indice j correspond à la direction en x dans l'image, l'indice i la direction en y , tandis que t est assimilé à l'axe des temps.

6.1.2.2 Estimation du Laplacien du flot des vitesses

A la vue des deux équations (6.9), il est nécessaire d'approximer les Laplaciens de w_x et w_y . L'approximation choisie par les deux auteurs prend la forme suivante :

$$\nabla^2 w_x \approx \kappa \cdot \left(\bar{w}_{(i,j)x} - w_{(i,j)x} \right) \quad \text{et} \quad \nabla^2 w_y \approx \kappa \cdot \left(\bar{w}_{(i,j)y} - w_{(i,j)y} \right) \quad (6.11)$$

où

$$\begin{aligned} \bar{w}_{(i,j)x} = & \frac{1}{6} \cdot \left(w_{(i-1,j)x} + w_{(i,j-1)x} + w_{(i,j+1)x} + w_{(i+1,j)x} \right) \\ & + \frac{1}{12} \cdot \left(w_{(i-1,j-1)x} + w_{(i-1,j+1)x} + w_{(i+1,j-1)x} + w_{(i+1,j+1)x} \right) \end{aligned} \quad (6.12a)$$

$$\begin{aligned} \bar{w}_{(i,j)y} = & \frac{1}{6} \cdot \left(w_{(i-1,j)y} + w_{(i,j-1)y} + w_{(i,j+1)y} + w_{(i+1,j)y} \right) \\ & + \frac{1}{12} \cdot \left(w_{(i-1,j-1)y} + w_{(i-1,j+1)y} + w_{(i+1,j-1)y} + w_{(i+1,j+1)y} \right) \end{aligned} \quad (6.12b)$$

La Figure 3 ci-dessous illustre l'approximation du Laplacien.

1/12	1/6	1/12
1/6	-1	1/6
1/12	1/6	1/12

Figure 3 : estimation du Laplacien

6.1.2.3 Solution itérative de la minimisation

La résolution de la paire d'équations (6.9), en ayant au préalable substitué les Laplaciens $\nabla^2 w_x$ et $\nabla^2 w_y$ par leurs approximations (6.11), s'effectue par une méthode itérative standard, en l'occurrence la méthode de Gauss-Seidel. Cette méthode peut être rapprochée au fonctionnement en mode asynchrone d'un réseau récursif. L'estimation de la nouvelle série de vitesse (w_x^{n+1}, w_y^{n+1}) à partir des valeurs précédentes (w_x^n, w_y^n) se déduit par :

$$w_x^{n+1} = \bar{w}_x^n - f_x \cdot (f_x \cdot \bar{w}_x^n + f_y \cdot \bar{w}_y^n + f_t) / (\alpha^2 + f_x^2 + f_y^2) \quad (6.13a)$$

$$w_y^{n+1} = \bar{w}_y^n - f_y \cdot (f_x \cdot \bar{w}_x^n + f_y \cdot \bar{w}_y^n + f_t) / (\alpha^2 + f_x^2 + f_y^2) \quad (6.13b)$$

Toutefois, ce problème correspond typiquement à celui des fonctionnelles minimisables par le biais de notre approche neuronale préalablement exposée au chapitre 5.

6.2 Méthodologie neuronale proposée pour le flot optique

La construction du réseau basé sur le concept de Hopfield qui doit répondre au problème de l'estimation du mouvement à travers l'approche du flot optique peut se décomposer en quatre étapes [Lelaurin, Yacoubi, Devlaminck, & Dubus, 1995] :

- le codage du vecteur vitesse $\vec{w} = (w_x, w_y)$
- l'architecture et la dynamique du réseau
- la fonction d'énergie du réseau
- la phase d'identification avec la fonctionnelle du flot optique

6.2.1 Codage du vecteur vitesse

La représentation du vecteur vitesse \vec{w} en un point (i,j) de l'image, ou plus précisément les deux composantes qui le constituent, se fait en termes d'états de neurones binaires comme décrit au chapitre 5 §5.2.3. Le modèle général est ici adapté au fait que l'on a deux grandeurs à estimer (w_x, w_y) , et que ces dernières sont dimensionnées à valeurs positives ou négatives. Ceci nécessite de décrire les variables à estimer par une sommation algébrique de neurones appartenant à deux sous-réseaux, un pour la partie positive et l'autre pour la partie négative. Soit au total quatre sous-réseaux en chaque point (i,j) de l'image afin d'estimer $w_{(i,j)x}$ et $w_{(i,j)y}$. Cette démarche revient à reformuler les deux composantes sous l'écriture suivante :

$$\begin{aligned}
 w_{(i,j)x} &= w_{(i,j)x}^+ - w_{(i,j)x}^- \\
 w_{(i,j)y} &= w_{(i,j)y}^+ - w_{(i,j)y}^-
 \end{aligned}
 \tag{6.14}$$

où

$$\begin{aligned}
 w_{(i,j)x}^+ &= k \cdot \sum_{h^+=1}^N v_{i,j,h^+} & \text{et} & & w_{(i,j)x}^- &= k \cdot \sum_{h^-=1}^N v_{i,j,h^-} \\
 w_{(i,j)y}^+ &= k \cdot \sum_{p^+=1}^N v_{i,j,p^+} & \text{et} & & w_{(i,j)y}^- &= k \cdot \sum_{p^-=1}^N v_{i,j,p^-}
 \end{aligned}
 \tag{6.15}$$

Le paramètre de quantification k est à valeur réelle positive et dépend de la séquence d'images à traiter. Il définit ainsi l'échelle de quantification au sein de la dynamique des vecteurs vitesse. Le produit $k \cdot N$, où N représente le nombre de neurones dans un sous-réseau (voir Figure 4), détermine alors la valeur maximale détectable de la vitesse. Rappelons que la méthode du flot optique, de par l'approximation des gradients spatio-temporels, ne peut estimer correctement que de faibles déplacements des objets présents dans l'image.

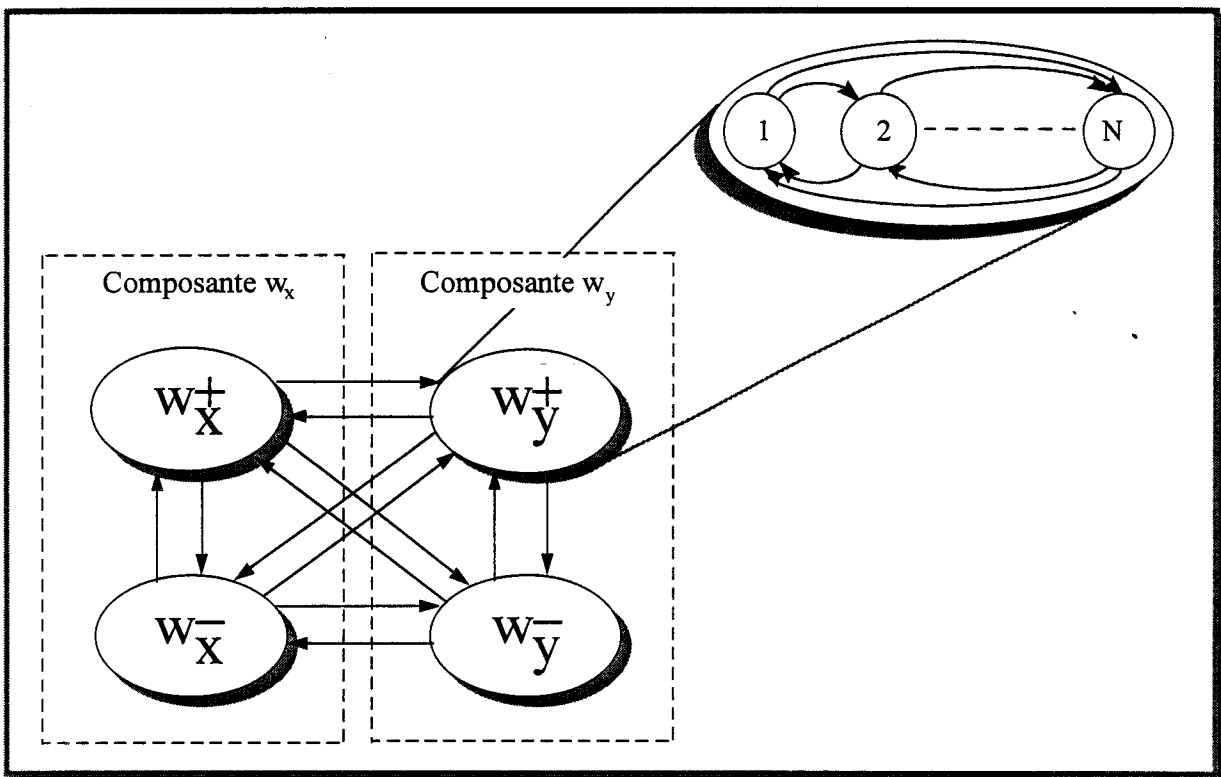


Figure 4 : Représentation du vecteur vitesse

Notons que pour les formules neuronales à suivre, les indices 'h' et 'l' seront associés aux sous-réseaux de la composante w_x , les indices 'p' et 'q' aux sous-réseaux de la

composante w_y , les indices '+' et '-' permettent quant à eux de distinguer respectivement les sous-réseaux des composantes à valeurs positives et négatives.

6.2.2 Architecture et dynamique du réseau pour le flot optique

6.2.2.1 Architecture globale du réseau

Le réseau global se constitue de $4 \times L \times M$ sous-réseaux de type Hopfield, ou encore $4 \times L \times M \times N$ neurones binaires, où $L \times M$ représentent les dimensions de la séquence d'images à traiter. La Figure 5 schématise partiellement l'architecture de ce réseau. Tous les sous-réseaux, en chaque point de l'image, sont connectés entre eux. Plus précisément, chaque neurone d'un sous-réseau possède une connexion avec tous les autres neurones à l'intérieur du réseau global et une connexion avec lui-même, ou auto-connexion. Un neurone compte $4 \times L \times M \times N$ connexions. En vérité, nous rappelons que l'horizon des connexions est limité et dépend de la différence finie due au terme de régularisation comme il l'a été explicité au chapitre 5 §5.2.4.

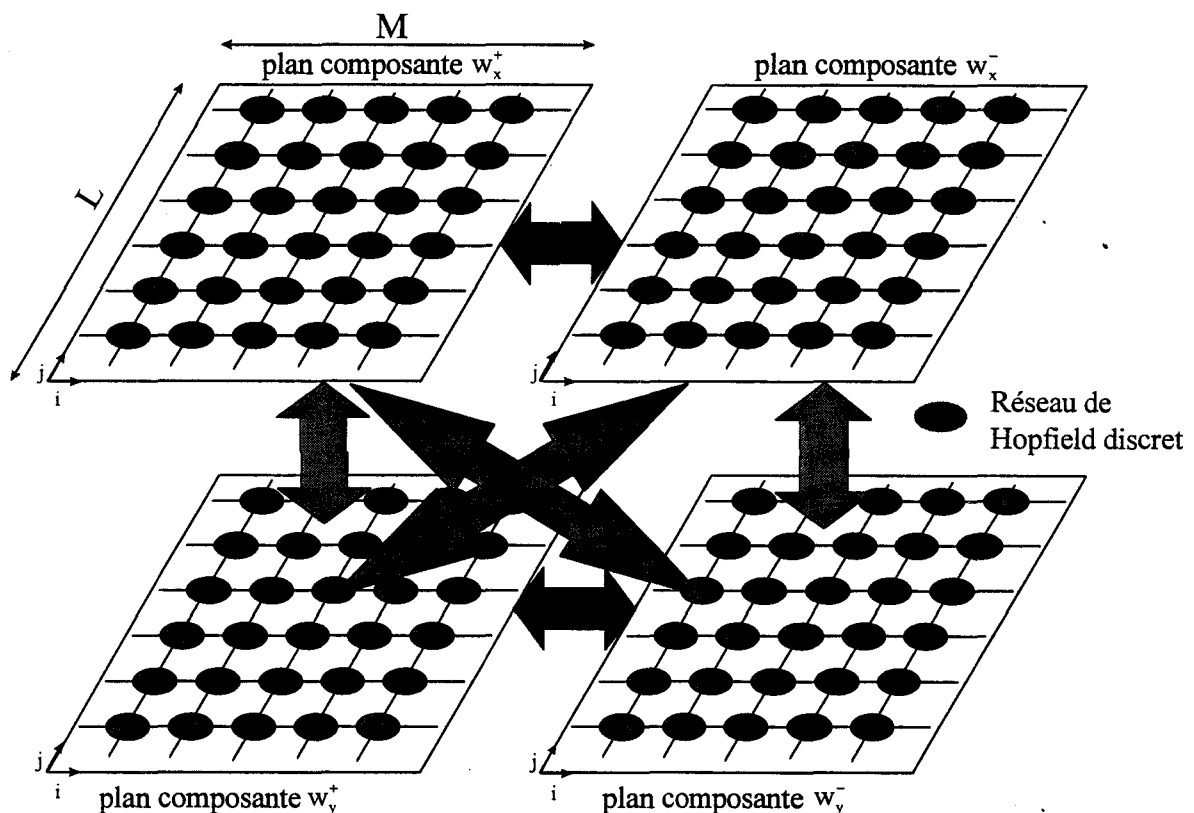


Figure 5 : Architecture globale du réseau pour le flot optique.
4 plans image constitués de $L \times M$ sous-réseaux totalement interconnectés

6.2.2.2 La dynamique

La dynamique du réseau est entièrement définie par la donnée de la nature des connexions, celle de la fonction d'activation des neurones et enfin par la définition du mode d'itération. La structure des connexions présente un grand intérêt vis-à-vis de la convergence du réseau. Dans notre cas, cette convergence est assurée par les poids de connexions symétriques et une mise à jour asynchrone dans laquelle le vecteur d'état est actualisé élément par élément en tenant compte des nouvelles valeurs des composantes dès qu'elles sont disponibles.

Pour chacun des quatre sous-réseaux, le potentiel $u_{i,j,\bullet}^1$ du neurone $(i, j, \bullet)^1$ se définit par les relations suivantes :

➤ *sous-réseaux de la composante x*

◆ valeurs positives

$$\begin{aligned} u_{i,j,h^+} = & \sum_{m=1}^L \sum_{n=1}^M \sum_{l^+=1}^N T_{i,j,h^+:m,n,l^+} \cdot V_{m,n,l^+} + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^-=1}^N T_{i,j,h^+:m,n,h^-} \cdot V_{m,n,h^-} \\ & + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^+=1}^N T_{i,j,h^+:m,n,p^+} \cdot V_{m,n,p^+} + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^-=1}^N T_{i,j,h^+:m,n,p^-} \cdot V_{m,n,p^-} + I_{i,j,h^+} \end{aligned} \quad (6.16a)$$

◆ valeurs négatives

$$\begin{aligned} u_{i,j,h^-} = & \sum_{m=1}^L \sum_{n=1}^M \sum_{l^-=1}^N T_{i,j,h^-:m,n,l^-} \cdot V_{m,n,l^-} + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^+=1}^N T_{i,j,h^-:m,n,h^+} \cdot V_{m,n,h^+} \\ & + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^-=1}^N T_{i,j,h^-:m,n,p^-} \cdot V_{m,n,p^-} + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^+=1}^N T_{i,j,h^-:m,n,p^+} \cdot V_{m,n,p^+} + I_{i,j,h^-} \end{aligned} \quad (6.16b)$$

➤ *sous-réseaux de la composante y*

◆ valeurs positives

$$\begin{aligned} u_{i,j,p^+} = & \sum_{m=1}^L \sum_{n=1}^M \sum_{q^+=1}^N T_{i,j,p^+:m,n,q^+} \cdot V_{m,n,q^+} + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^-=1}^N T_{i,j,p^+:m,n,p^-} \cdot V_{m,n,p^-} \\ & + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^+=1}^N T_{i,j,p^+:m,n,h^+} \cdot V_{m,n,h^+} + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^-=1}^N T_{i,j,p^+:m,n,h^-} \cdot V_{m,n,h^-} + I_{i,j,p^+} \end{aligned} \quad (6.17a)$$

¹ Le symbole • remplace tous les indices $h^+, l^+, p^+, q^+, h^-, \dots$

♦ valeurs négatives

$$\begin{aligned}
 u_{i,j,p^-} = & \sum_{m=1}^L \sum_{n=1}^M \sum_{q^- = 1}^N T_{i,j,p^-:m,n,q^-} \cdot V_{m,n,q^-} + \sum_{m=1}^L \sum_{n=1}^M \sum_{p^+ = 1}^N T_{i,j,p^-:m,n,p^+} \cdot V_{m,n,p^+} \\
 & + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^- = 1}^N T_{i,j,p^-:m,n,h^-} \cdot V_{m,n,h^-} + \sum_{m=1}^L \sum_{n=1}^M \sum_{h^+ = 1}^N T_{i,j,p^-:m,n,h^+} \cdot V_{m,n,h^+} + I_{i,j,p^-}
 \end{aligned}
 \tag{6.17b}$$

Dans le cas d'un neurone h^+ , $T_{i,j,h^+:m,n,l^+}$ représente les interconnexions synaptiques avec les neurones des sous-réseaux $w_{(i,j)x}^+$. Les interconnexions $T_{i,j,h^+:m,n,h^-}$ lient ce même neurone avec ceux des sous-réseaux $w_{(i,j)x}^-$. $T_{i,j,h^+:m,n,p^+}$ et $T_{i,j,h^+:m,n,p^-}$ symbolisant la connexion avec les neurones des deux sous-réseaux de la composante ω_y . Enfin, I_{i,j,h^+} est l'entrée externe du neurone h^+ . De même, les autres $T_{i,j,*,m,n,*}$ caractérisent les poids synaptiques reliant chaque neurone des trois autres sous-réseaux à l'ensemble des neurones du réseau global et $I_{i,j,*}$, leurs entrées externes. La Figure 6 ci-dessous présente ces différentes connexions et l'entrée externe.

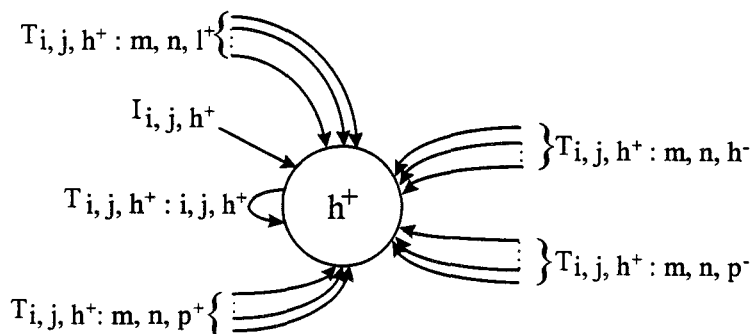


Figure 6 : modèle de connexions d'un neurone h^+

L'état $V_{i,j,*}^l$ de chaque neurone est défini par une fonction d'activation :

$$V_{i,j,h^+} = g(u_{i,j,h^+}) \tag{6.18a}$$

$$V_{i,j,h^-} = g(u_{i,j,h^-}) \tag{6.18b}$$

$$V_{i,j,p^+} = g(u_{i,j,p^+}) \tag{6.18c}$$

$$V_{i,j,p^-} = g(u_{i,j,p^-}) \tag{6.18d}$$

où $g(x)$ est la fonction de Heaviside dont le rôle est prépondérant dans le comportement du réseau. A titre de rappel, cette fonction se définit comme suit :

$$g(x) = \begin{cases} 1 & \text{si } x > 0 \\ 0 & \text{si } x < 0 \end{cases} \quad (6.19)$$

6.2.2.3 La fonction d'énergie

Notre architecture neuronale est d'un concept similaire au modèle de Hopfield [Hopfield & Tank, 1986]. Nous avons en effet quatre sous-réseaux $w_{(i,j)x}^+$, $w_{(i,j)x}^-$, $w_{(i,j)y}^+$ et $w_{(i,j)y}^-$ de type Hopfield discret chaque point (i, j) de l'image. Leur évolution dynamique se caractérise donc respectivement par une fonction d'énergie, que l'on dénommera par la suite sous le terme de *sous-énergie* pour la distinguer de l'énergie globale du réseau. Chacune de ces quatre sous-énergies $E_{(i,j)x}^+$, $E_{(i,j)x}^-$, $E_{(i,j)y}^+$ et $E_{(i,j)y}^-$ ont pour expression :

➤ *sous-réseaux de composante x*

◆ valeurs positives

$$E_{(i,j)x}^+ = -\frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{h^+=1}^N \left[\sum_{l^+=1}^N T_{i,j,h^+:m,n,l^+} \cdot V_{m,n,l^+} \cdot V_{i,j,h^+} + \sum_{h^-=1}^N T_{i,j,h^+:m,n,h^-} \cdot V_{m,n,h^-} \cdot V_{i,j,h^+} \right. \\ \left. + \sum_{p^+=1}^N T_{i,j,h^+:m,n,p^+} \cdot V_{m,n,p^+} \cdot V_{i,j,h^+} + \sum_{p^-=1}^N T_{i,j,h^+:m,n,p^-} \cdot V_{m,n,p^-} \cdot V_{i,j,h^+} \right] - \sum_{h^+=1}^N I_{i,j,h^+} \cdot V_{i,j,h^+} \quad (6.20a)$$

◆ valeurs négatives

$$E_{(i,j)x}^- = -\frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{h^-=1}^N \left[\sum_{l^-=1}^N T_{i,j,h^ -:m,n,l^-} \cdot V_{m,n,l^-} \cdot V_{i,j,h^-} + \sum_{h^+=1}^N T_{i,j,h^ -:m,n,h^+} \cdot V_{m,n,h^+} \cdot V_{i,j,h^-} \right. \\ \left. + \sum_{p^+=1}^N T_{i,j,h^ -:m,n,p^+} \cdot V_{m,n,p^+} \cdot V_{i,j,h^-} + \sum_{p^-=1}^N T_{i,j,h^ -:m,n,p^-} \cdot V_{m,n,p^-} \cdot V_{i,j,h^-} \right] - \sum_{h^-=1}^N I_{i,j,h^-} \cdot V_{i,j,h^-} \quad (6.20b)$$

➤ *sous-réseaux de la composante y*

◆ valeurs positives

$$E_{(i,j)y}^+ = -\frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{p^+=1}^N \left[\sum_{q^+=1}^N T_{i,j,p^+:m,n,q^+} \cdot V_{m,n,q^+} \cdot V_{i,j,p^+} + \sum_{p^+=1}^N T_{i,j,p^+:m,n,q^+} \cdot V_{m,n,q^+} \cdot V_{i,j,p^+} \right. \\ \left. + \sum_{h^+=1}^N T_{i,j,p^+:m,n,h^+} \cdot V_{m,n,h^+} \cdot V_{i,j,p^+} + \sum_{h^+=1}^N T_{i,j,p^+:m,n,h^+} \cdot V_{m,n,h^+} \cdot V_{i,j,p^+} \right] - \sum_{p^+=1}^N I_{i,j,p^+} \cdot V_{i,j,p^+} \quad (6.20c)$$

◆ valeurs négatives

$$E_{(i,j)y}^- = -\frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{p^-=1}^N \left[\sum_{q^-=1}^N T_{i,j,p^ -:m,n,q^-} \cdot V_{m,n,q^-} \cdot V_{i,j,p^-} + \sum_{p^-=1}^N T_{i,j,p^ -:m,n,p^-} \cdot V_{m,n,p^-} \cdot V_{i,j,p^-} \right. \\ \left. + \sum_{h^-=1}^N T_{i,j,p^ -:m,n,h^-} \cdot V_{m,n,h^-} \cdot V_{i,j,p^-} + \sum_{h^-=1}^N T_{i,j,p^ -:m,n,h^-} \cdot V_{m,n,h^-} \cdot V_{i,j,p^-} \right] - \sum_{p^-=1}^N I_{i,j,p^-} \cdot V_{i,j,p^-} \quad (6.20d)$$

La fonction d'énergie globale de ce réseau se déduit par une simple somme sur l'ensemble des points de l'image des quatre *sous-énergies* précédentes :

$$E = \sum_{i=1}^L \sum_{j=1}^M (E_{(i,j)x}^+ + E_{(i,j)x}^- + E_{(i,j)y}^+ + E_{(i,j)y}^-) \quad (6.21)$$

6.2.3 La phase d'identification

Au préalable, nous allons adapter la fonctionnelle originelle du flot optique (6.8) en la pondérant par le terme γ^2 , où γ est une constante inversement proportionnelle à un gradient spatial, pour obtenir la fonctionnelle à minimiser :

$$E = \sum_{i=1}^L \sum_{j=1}^M \gamma^2 \cdot \left\{ (f_x \cdot w_x + f_y \cdot w_y + f_t)^2 + \alpha^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_x}{\partial y} \right)^2 + \left(\frac{\partial w_y}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 \right] \right\} \quad (6.22)$$

L'intérêt de ce coefficient sera plus explicite après la phase d'identification. Il permet d'obtenir une homogénéité entre les signaux d'entrées des neurones et leurs signaux de sorties et est dû au fait que les variables sont dimensionnées. L'approximation des gradients spatio-temporels de la fonction intensité f est identique à celle décrite par les relations (6.10). Quant aux gradients spatiaux de la vitesse, leur approximation dans un domaine de dimensions 2×2 est, sous une écriture matricielle :

$$\frac{\partial}{\partial \mathbf{x}} \approx \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad \frac{\partial}{\partial \mathbf{y}} \approx \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} \quad (6.23)$$

Sous une formulation plus appropriée à sa transcription dans la relation (6.22) en s'appuyant sur le §5.2.2 du chapitre précédent, on réécrit ces deux expressions matricielles comme suit :

$$\frac{\partial}{\partial \mathbf{x}} = \sum_{a=0}^1 \sum_{b=0}^1 d_{a,b}^x \quad \text{où} \quad d_{a,b}^x = \begin{cases} -\frac{1}{2} & \text{si } a = 0 \\ \frac{1}{2} & \text{si } a = 1 \end{cases} \quad (6.24a)$$

$$\frac{\partial}{\partial \mathbf{y}} = \sum_{a=0}^1 \sum_{b=0}^1 d_{a,b}^y \quad \text{où} \quad d_{a,b}^y = \begin{cases} -\frac{1}{2} & \text{si } b = 0 \\ \frac{1}{2} & \text{si } b = 1 \end{cases} \quad (6.24b)$$

Compte tenu des résultats sur le modèle général décrit au chapitre 5 et de l'adaptation de ce modèle au cas de la fonctionnelle (6.22) que nous venons d'exposer, la phase d'identification des coefficients synaptiques $T_{i,j,m,n}$ et entrées externes $I_{i,j}$ conduit au résultat suivant :

➤ *Poids synaptiques*

- ◆ Pour les sous-réseaux $w_{(i,j)x}^+$ et $w_{(i,j)x}^-$

- Connexions intra sous-réseaux

$$T_{i,j,h^+,m,n,l^+} = T_{i,j,h^-,m,n,l^-} = -2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)x}^2 + 2 \cdot \alpha^2 \right) \cdot \delta_{(i,j),(m,n)} - \frac{1}{2} \cdot \alpha^2 \cdot \delta_{(i,j),(m,n) \oplus s} \right] \quad (6.25a)$$

- Connexions inter sous-réseaux $w_{(i,j)x}^+ \rightarrow w_{(i,j)x}^-$

$$T_{i,j,h^+,m,n,h^-} = T_{i,j,h^-,m,n,h^+} = +2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)x}^2 + 2 \cdot \alpha^2 \right) \cdot \delta_{(i,j),(m,n)} - \frac{1}{2} \cdot \alpha^2 \cdot \delta_{(i,j),(m,n) \oplus s} \right] \quad (6.25b)$$

- Connexions inter composantes $w_{(i,j)x} \leftrightarrow w_{(i,j)y}$

* $w_{(i,j)x}^+ \rightarrow w_{(i,j)y}^+$ et $w_{(i,j)x}^- \rightarrow w_{(i,j)y}^-$

$$T_{i,j,h^+;m,n,p^+} = T_{i,j,h^-;m,n,p^-} = -2 \cdot \gamma^2 \cdot k^2 \cdot f_{(i,j)x} \cdot f_{(i,j)y} \cdot \delta_{(i,j),(m,n)} \quad (6.25c)$$

* $w_{(i,j)x}^+ \rightarrow w_{(i,j)y}^-$ et $w_{(i,j)x}^- \rightarrow w_{(i,j)y}^+$

$$T_{i,j,h^+;m,n,p^-} = T_{i,j,h^-;m,n,p^+} = +2 \cdot \gamma^2 \cdot k^2 \cdot f_{(i,j)x} \cdot f_{(i,j)y} \cdot \delta_{(i,j),(m,n)} \quad (6.25d)$$

- ◆ Pour les sous-réseaux $w_{(i,j)y}^+$ et $w_{(i,j)y}^-$

- Connexions intra sous-réseaux

$$T_{i,j,p^+;m,n,q^+} = T_{i,j,p^-;m,n,q^-} = -2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)y}^2 + 2 \cdot \alpha^2 \right) \cdot \delta_{(i,j),(m,n)} - \frac{1}{2} \cdot \alpha^2 \cdot \delta_{(i,j),(m,n) \oplus s} \right] \quad (6.25c)$$

- Connexions inter sous-réseaux $w_{(i,j)y}^+ \leftrightarrow w_{(i,j)y}^-$

$$T_{i,j,p^+;m,n,p^-} = T_{i,j,p^-;m,n,p^+} = +2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)y}^2 + 2 \cdot \alpha^2 \right) \cdot \delta_{(i,j),(m,n)} - \frac{1}{2} \cdot \alpha^2 \cdot \delta_{(i,j),(m,n) \oplus s} \right] \quad (6.25d)$$

- Connexions inter composantes $w_{(i,j)y} \rightarrow w_{(i,j)x}$

* $w_{(i,j)y}^+ \rightarrow w_{(i,j)x}^+$ et $w_{(i,j)y}^- \rightarrow w_{(i,j)x}^-$

$$T_{i,j,p^+;m,n,h^+} = T_{i,j,p^-;m,n,h^-} = -2 \cdot \gamma^2 \cdot k^2 \cdot f_{(i,j)x} \cdot f_{(i,j)y} \cdot \delta_{(i,j),(m,n)} \quad (6.25e)$$

* $w_{(i,j)y}^- \rightarrow w_{(i,j)x}^+$ et $w_{(i,j)y}^+ \rightarrow w_{(i,j)x}^-$

$$T_{i,j,p^+;m,n,h^-} = T_{i,j,p^-;m,n,h^+} = +2 \cdot \gamma^2 \cdot k^2 \cdot f_{(i,j)x} \cdot f_{(i,j)y} \cdot \delta_{(i,j),(m,n)} \quad (6.25f)$$

➤ *entrées externes*

- ◆ Pour les sous-réseaux $w_{(i,j)x}^+$ et $w_{(i,j)x}^-$

$$I_{i,j,h^+} = -2 \cdot \gamma^2 \cdot k \cdot f_{(i,j)x} \cdot f_{(i,j)t} \quad \text{et} \quad I_{i,j,h^-} = -I_{i,j,h^+} \quad (6.26a)$$

- ◆ Pour les sous-réseaux $w_{(i,j)y}^+$ et $w_{(i,j)y}^-$

$$I_{i,j,p^+} = -2 \cdot \gamma^2 \cdot k \cdot f_{(i,j)y} \cdot f_{(i,j)t} \quad \text{et} \quad I_{i,j,p^-} = -I_{i,j,p^+} \quad (6.26b)$$

où $\delta_{a,b}$ est la fonction de Kronecker, $S = \{(-1, -1), (-1, 1), (1, -1), (1, 1)\}$ est une série d'index. A titre de rappel, le symbole \oplus signifie :

$$f_{a \oplus b} = \begin{cases} f_{a+b} & \text{si } 0 \leq a + b \leq L, M \\ 0 & \text{ailleurs} \end{cases} \quad (6.27)$$

L'utilisation du réseau consiste à forcer l'état initial des neurones et à laisser évoluer la structure bouclée vers un état d'équilibre qui fournit le résultat du traitement neuronal. A cet effet, une initialisation assez proche de la solution permettra d'éviter de tomber dans un mauvais minimum local.

Nous vérifions que la matrice des poids synaptiques est symétrique, ce qui augure de la convergence du réseau. Il est à noter par ailleurs que ces poids synaptiques n'évoluent pas durant le processus de convergence du réseau car ils sont complètement spécifiés par l'énergie à minimiser.

Si on analyse les différentes formulations relatives aux poids synaptiques, nous remarquons que le coefficient γ est un facteur pour chacune d'entre-elles. Si on omet ce paramètre, on constate que les poids synaptiques $T_{i,j,\bullet,m,n,\bullet}^1$ possède une notion de dimension (proportionnels à un gradient spatial). Or, d'après J. Hopfield [Hopfield, 1985], il est nécessaire que ces coefficients soient sans dimensions ce qui correspond à avoir une fonction de Heaviside avec une entrée et une sortie de même nature (relation $u_i - V_i$). La solution implicite serait de diviser les poids synaptiques par le gradient spatial en chaque point (i,j) de l'image. Cette idée est cependant à proscrire puisque la symétrie des poids synaptiques ne serait plus respectée d'où une non-convergence du réseau. En conséquence, nous prenons comme valeur de γ , le gradient spatial moyen de la fonction intensité de l'image :

$$\gamma = \left(\frac{\sqrt{\sum_{i=1}^L \sum_{j=1}^M (f_{(i,j)x}^2 + f_{(i,j)y}^2)}}}{L \cdot M} \right)^{-1} \quad (6.28)$$

De plus, nous vérifions que la source externe d'informations $I_{i,j}$ ¹ du réseau est relative à une vitesse et si l'on transcrit le potentiel d'un neurone h^+ , en remplaçant les poids synaptiques et l'entrée externe par leurs expressions respectives, on a :

$$u_{i,j,h^+} = -2 \cdot \gamma^2 \cdot k \cdot f_{(i,j)x} \cdot \left[f_{(i,j)x} \cdot w_{(i,j)x} + f_{(i,j)y} \cdot w_{(i,j)y} + f_{(i,j)t} \right] + \alpha^2 \cdot k \cdot \left[w_{(i-1,j-1)x} + w_{(i-1,j+1)x} - 4 \cdot w_{(i,j)x} + w_{(i+1,j-1)x} + w_{(i+1,j+1)x} \right] \quad (6.29)$$

Au vue de cet équation, on constate que le potentiel d'un neurone est bien proportionnelle à une vitesse, donc l'homogénéité entrée-sortie du neurone est vérifiée.

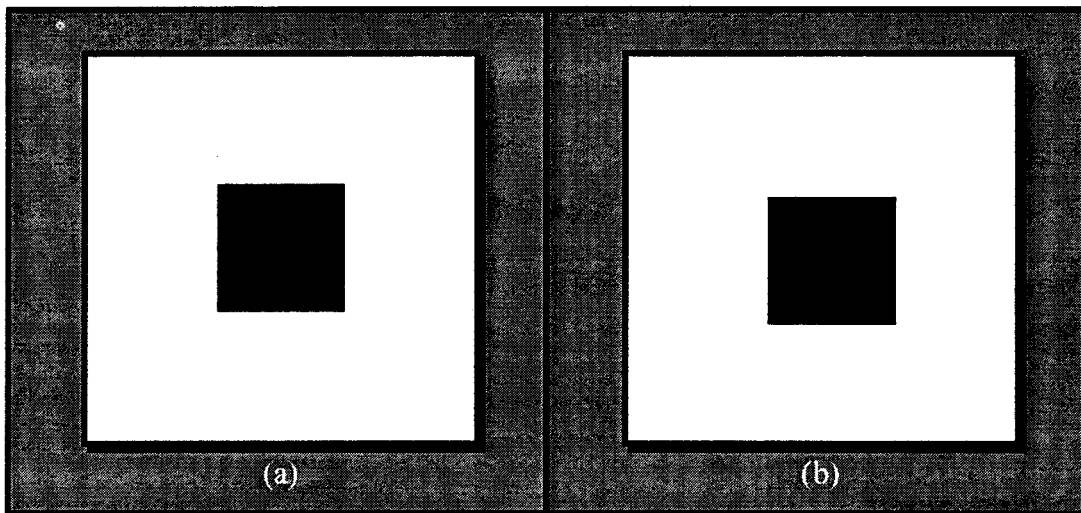
6.3 Résultats expérimentaux

Les performances du réseau proposé sont évaluées sur des images de synthèse. Des résultats sur des images réelles sont également donnés afin de montrer les avantages et les limites de notre approche.

6.3.1 Séquence d'images de synthèse

Afin de valider notre approche neuronale et d'effectuer la comparaison avec l'approche classique de Horn & Shunck, deux séquences d'images de synthèse ont été générées. La Séquence 1 est constituée d'un carré noir de dimensions 10×10 sur un fond blanc subissant un mouvement translationnel de 1 pixel. La Séquence 2 fondée sur le même carré synthétise un mouvement rotationnel de 2° de ce dernier. Les différentes cartes de champs de vitesses représentées se limitent à la seule zone du carré en mouvement.

6.3.1.1 Cas de la translation pure



Séquence 1 : Carré noir en translation

Dans un premier temps, nous estimons le champ des vecteurs vitesses par l'approche de Horn & Schunk. Une valeur de 100 est affectée au paramètre de régularisation α afin d'obtenir un champ dense et de compenser le manque d'information au sein du carré (gradients spatio-temporels nuls). La Figure 7 ci-dessous présente la carte des vecteurs ainsi obtenus.

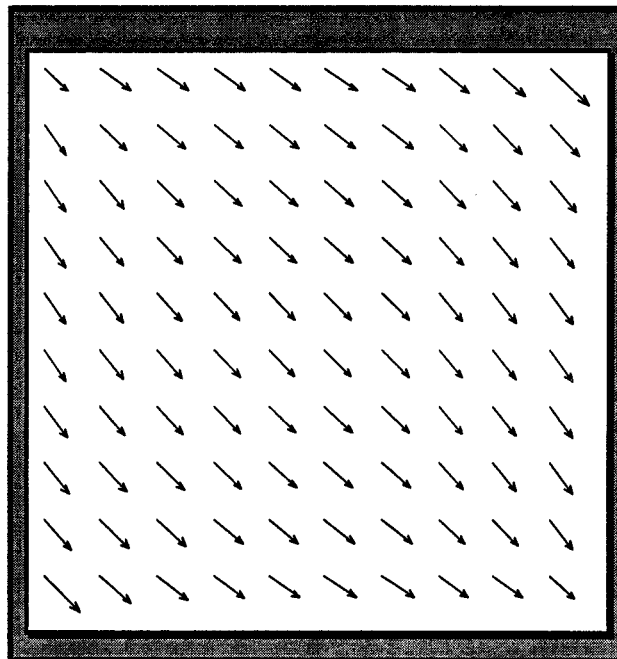


Figure 7 : Champ des vitesses par Horn & Schunk

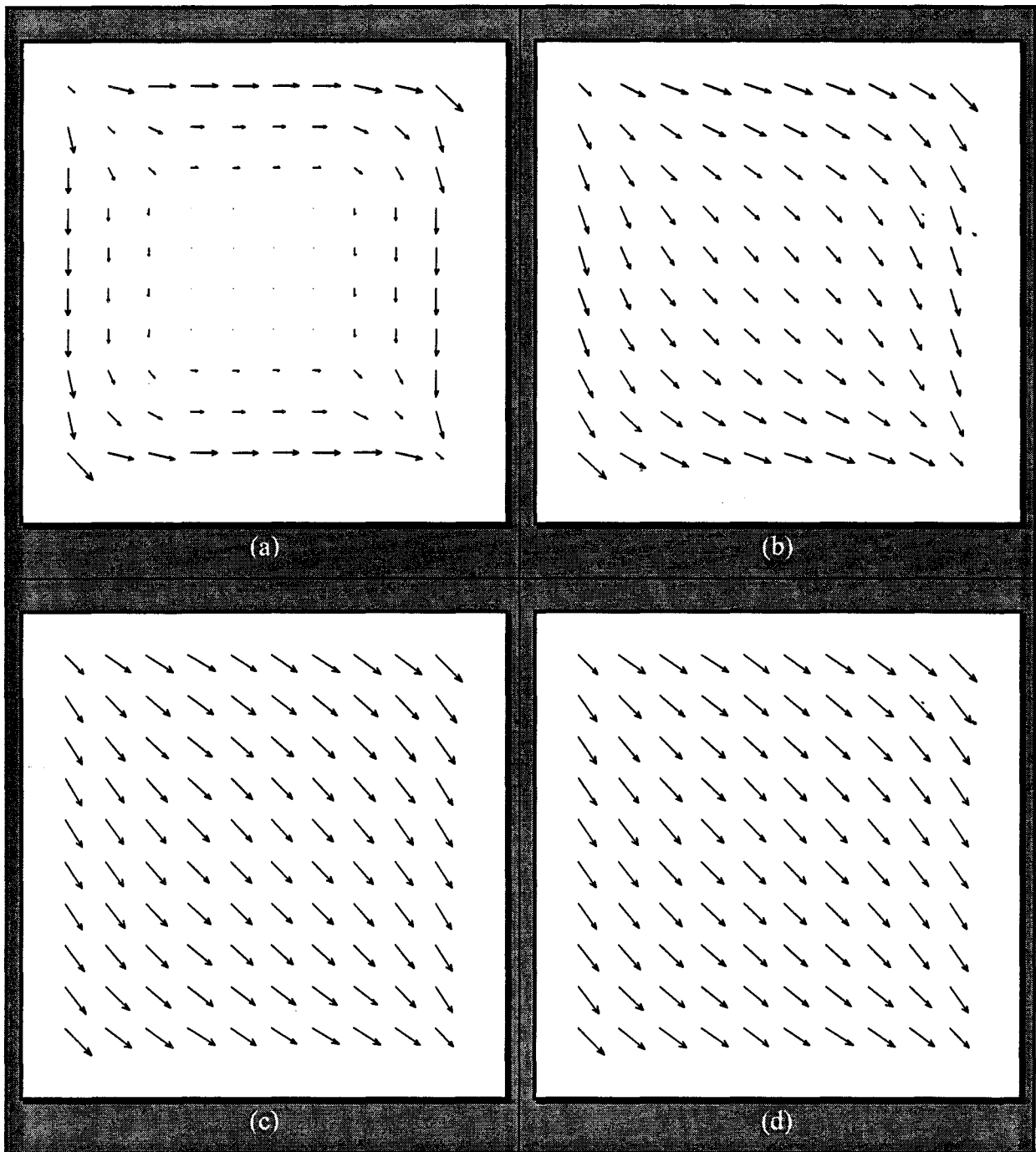


Figure 8 : Champs des vitesses pour différentes valeurs de N
(a) $N=5$, (b) $N=20$, (c) $N=100$, (d) $N=200$

Pour une même valeur du paramètre de régularisation α égale 100, nous fixons un nombre N de neurones donné et nous laissons le système neuronal évoluer jusqu'à l'obtention d'une valeur minimale de l'énergie finale, un état stable du réseau. Des tests pour $N=5$, $N=20$, $N=100$ et $N=200$ ont ainsi été réalisés. La dynamique des vitesses, traduite par le produit $k \cdot N$, pour chacune des quatre configurations est identique et égale à 1. Autrement dit, le pas de quantification k vaut respectivement 0.2, 0.05, 0.01 et 0.005.

Les champs de vecteurs vitesses, Figure 8, sont obtenus respectivement après 6, 20, 47 et 57 itérations. Nous remarquons, lorsque $N=5$, que la carte des vecteurs vitesses est incomplète et présente des vecteurs aberrants sur les bords. Au contraire, les champs obtenus pour des valeurs de N plus importantes sont de plus en plus denses et cohérents au fur et à mesure que le nombre de neurones augmente ($N=20$, $N=100$, puis $N=200$). La régularité de la solution augmente avec l'ordre du nombre de neurones. L'explication de ce résultat est implicite puisque l'augmentation du nombre de neurones coïncide avec un pas de quantification k de plus en plus faible.

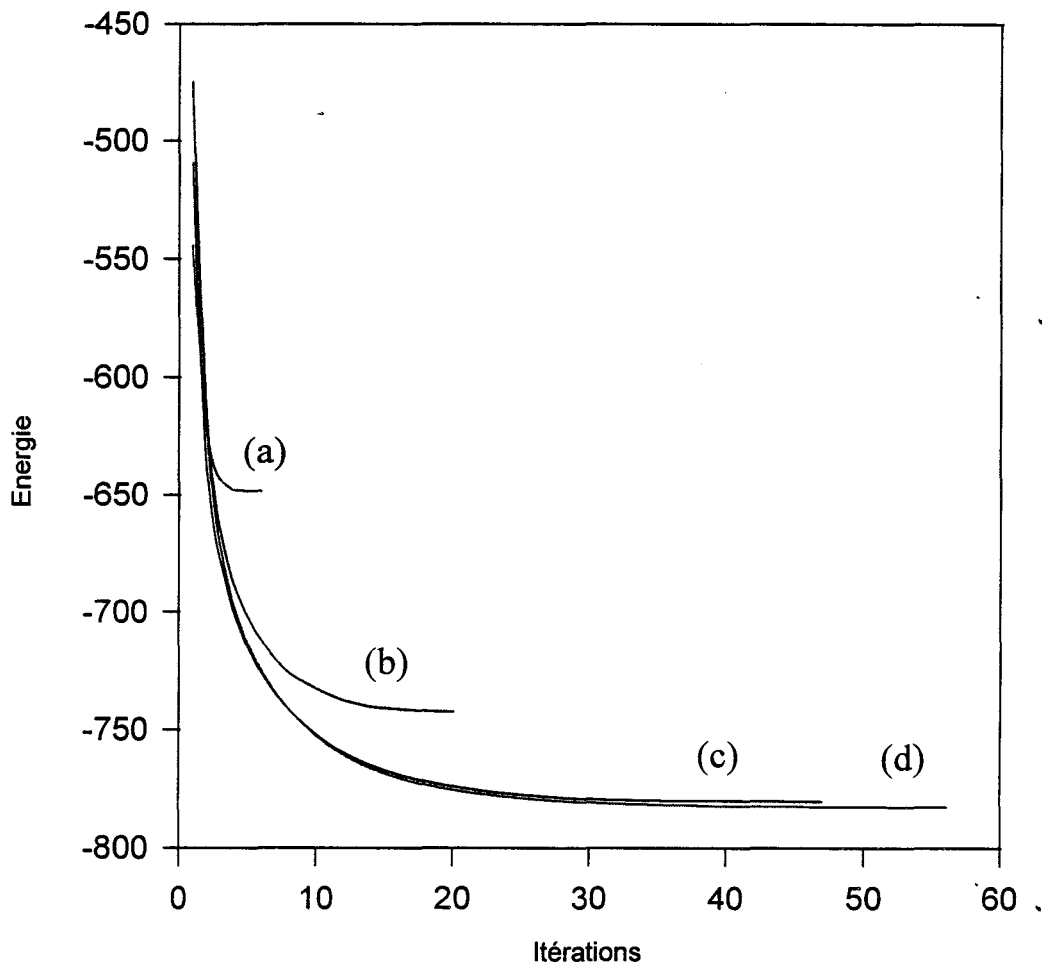


Figure 9 : Evolution de l'énergie du réseau dans les 4 configurations
(a) $N=5$, (b) $N=20$, (c) $N=100$, (d) $N=200$

L'amélioration de la qualité de la solution se constate aussi sur l'évolution temporelle des courbes d'énergies du réseau et la valeur finale de l'énergie (minimum local de la solution), Figure 9. Le minimum local où l'énergie est la plus faible correspond à un nombre

de neurones égal à 200. On peut toutefois se contenter de la configuration $N=100$ puisque, comparée à $N=200$, la solution finale en est très proche pour un nombre d'itération inférieure.

A titre de comparaison le Tableau 1 présente les erreurs moyennes en amplitude et en direction des différentes approches. Ces erreurs sont calculées sur les relations suivantes :

$$\varepsilon_{\text{amp}} (\%) = \frac{\sum_{i=1}^{L'} \sum_{j=1}^{M'} \left| \sqrt{2} - \sqrt{w_{(i,j)x}^2 + w_{(i,j)y}^2} \right|}{L' * M' * \sqrt{2}} * 100 \quad (6.29)$$

$$\varepsilon_{\text{ang}} (^\circ) = \frac{\sum_{i=1}^{L'} \sum_{j=1}^{M'} \left\{ 45 - \arctg \left(w_{(i,j)y} / w_{(i,j)x} \right) \right\}}{L' * M'}$$

Le calcul s'est effectué sur la zone du carré et non sur l'ensemble de l'image d'où $L'=M'=10$.

	Horn & schunk	N=5	N=20	N=100	N=200
Erreur en amplitude (%)	14.79	67.4097	39.2024	21.9799	20.2415
Erreur en direction (°)	4.4381	39.9252	14.1798	6.3564	5.3683

Tableau 1 : Erreur moyenne d'amplitude et de direction sur la zone du carré

Ces résultats confirment les observations précédentes puisque l'on peut noter que les erreurs sont d'autant plus importantes que le nombre de neurones diminue.

De plus, nous pouvons noter que ces courbes présentent deux zones distinctes de variation. Dans un premier temps, l'énergie décroît fortement puis la variation d'énergie diminue. La zone de faible variation correspond à une très faible amélioration de la qualité de la solution. Cette constatation permet d'imaginer un asservissement de la durée du temps de calcul (arrêt anticipé des itérations) sur le gradient de la courbe d'énergie. Le critère de terminaison préalable des itérations étant lié à un seuil de ce gradient.

La Figure 10 présente les différentes courbes de la dérivée temporelle de la fonction d'énergie pour les quatre configurations précédentes.

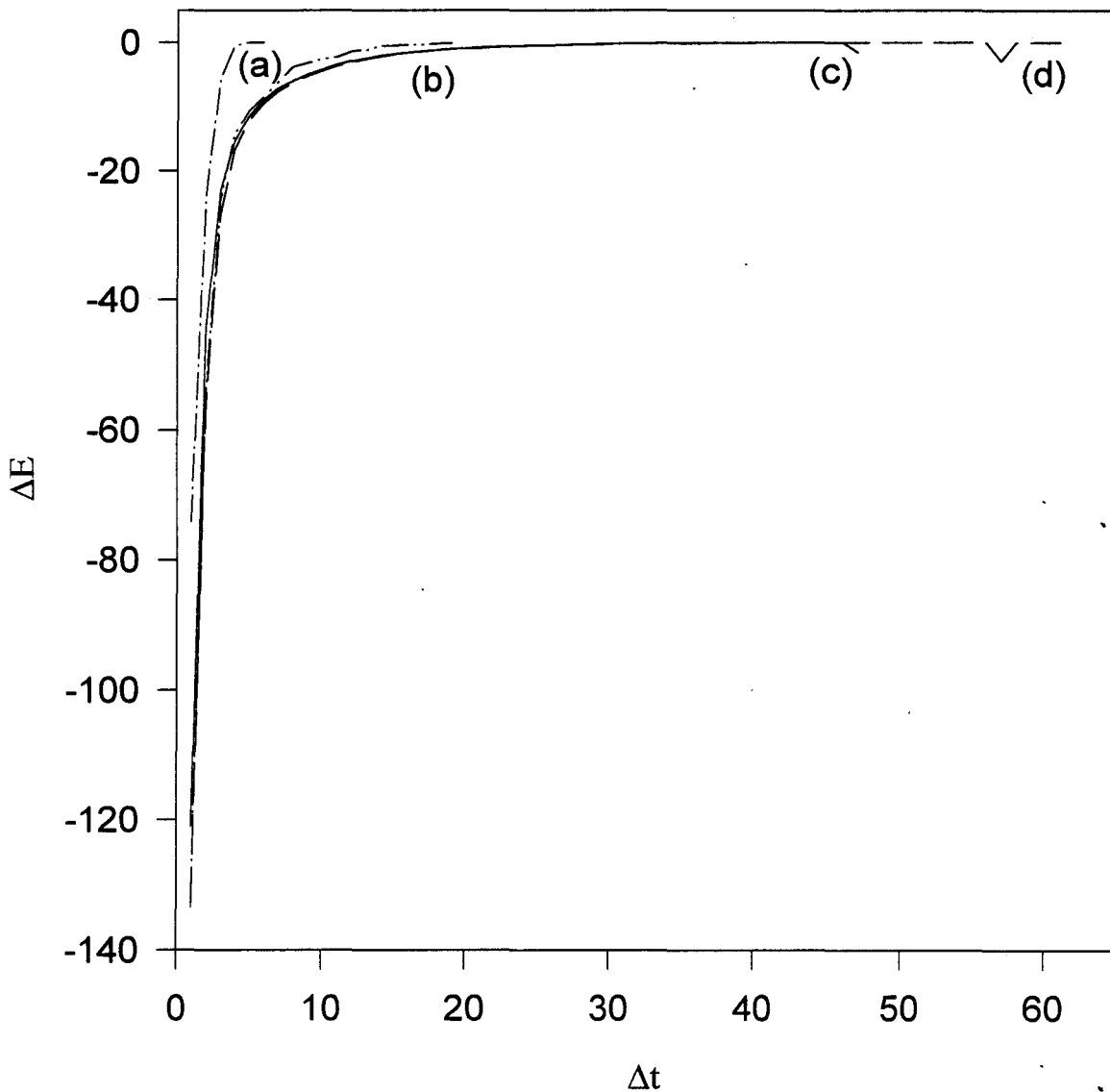


Figure 10 : Dérivée temporelle de la fonction d'énergie pour
(a) $N=5$, (b) $N=20$, (c) $N=100$, (d) $N=200$

Le Tableau 2 présente les erreurs en amplitude et direction obtenues pour une configuration de réseaux avec $N=100$, et pour quatre valeurs de seuil de gradient d'énergie différentes ($\Delta E < 5$, $\Delta E < 1$, $\Delta E < 0.1$, $\Delta E < 0.05$). Le réseau converge respectivement après 9, 19, 32 et 37 itérations. Si on compare les valeurs de ce tableau à celles du Tableau 1 pour $N=100$, nous constatons que pour un seuil de 0.05, les erreurs sont très proches. La Figure 11.b

confirme, sur la carte des vecteurs vitesses, la qualité de la solution finale obtenue avec une diminution de 20% du temps de calcul.

Seuil du gradient	$\Delta E < 5$	$\Delta E < 1$	$\Delta E < 0.1$	$\Delta E < 0.05$
Erreur en amplitude (%)	36.0622	27.6221	25.8076	21.9973
Erreur en direction (°)	15.1892	11.9918	10.5593	6.3664

Tableau 2 : Erreur moyenne en amplitude et en direction pour 4 valeurs de seuil de gradient

La Figure 11 montre la carte des champs de vitesses obtenue pour deux valeurs de seuil du gradient de l'énergie ($\Delta E < 1$, $\Delta E < 0.05$). Nous vérifions pour le premier cas, Figure 11.a, des erreurs sur certains vecteurs et dans le second cas, Figure 11.b, un champ tout à fait satisfaisant.

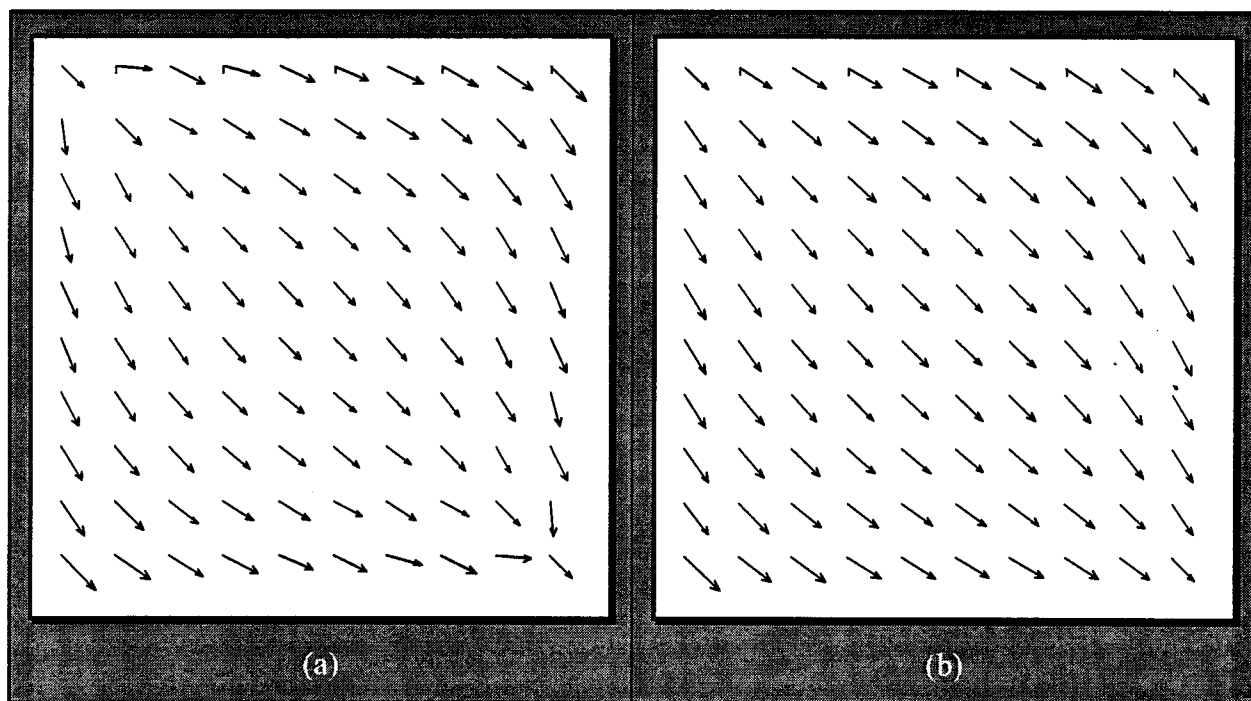
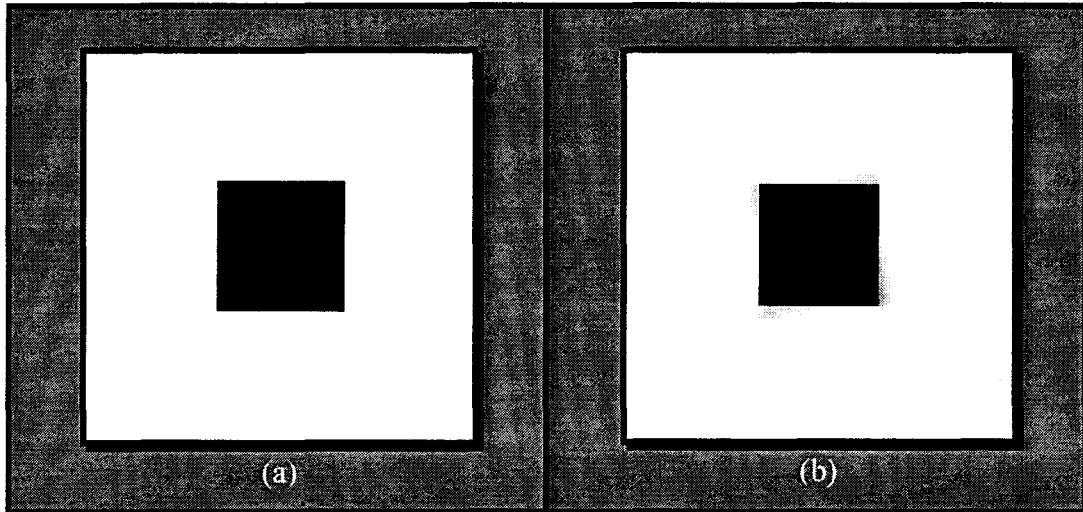


Figure 11 : Champ de vitesses pour $N=100$

6.3.1.2 cas de la rotation

De façon analogue, nous avons testé l'architecture neuronale dans le cas d'un mouvement rotationnel de ce même carré, Séquence 2. Le champ de vitesses est le résultat d'une rotation d'un angle de 2° autour de l'axe optique de la caméra.



Séquence 2 : carré noir en rotation

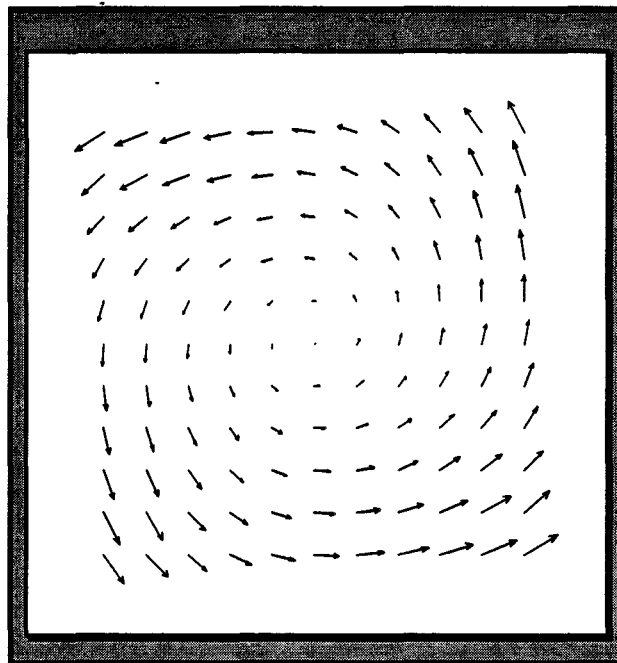


Figure 12 : Champ des vitesses par Horn & Schunk

De même que pour la translation, nous remarquons qu'une bonne solution du champ rotationnel impose un nombre de neurones suffisants, Figure 13. Dans ces conditions $N=200$, l'énergie est plus minimale, Figure 14. La Figure 12 montre le résultat obtenu avec l'algorithme de Horn & Schunk. Le coefficient de régularisation α est toujours fixé à 100:

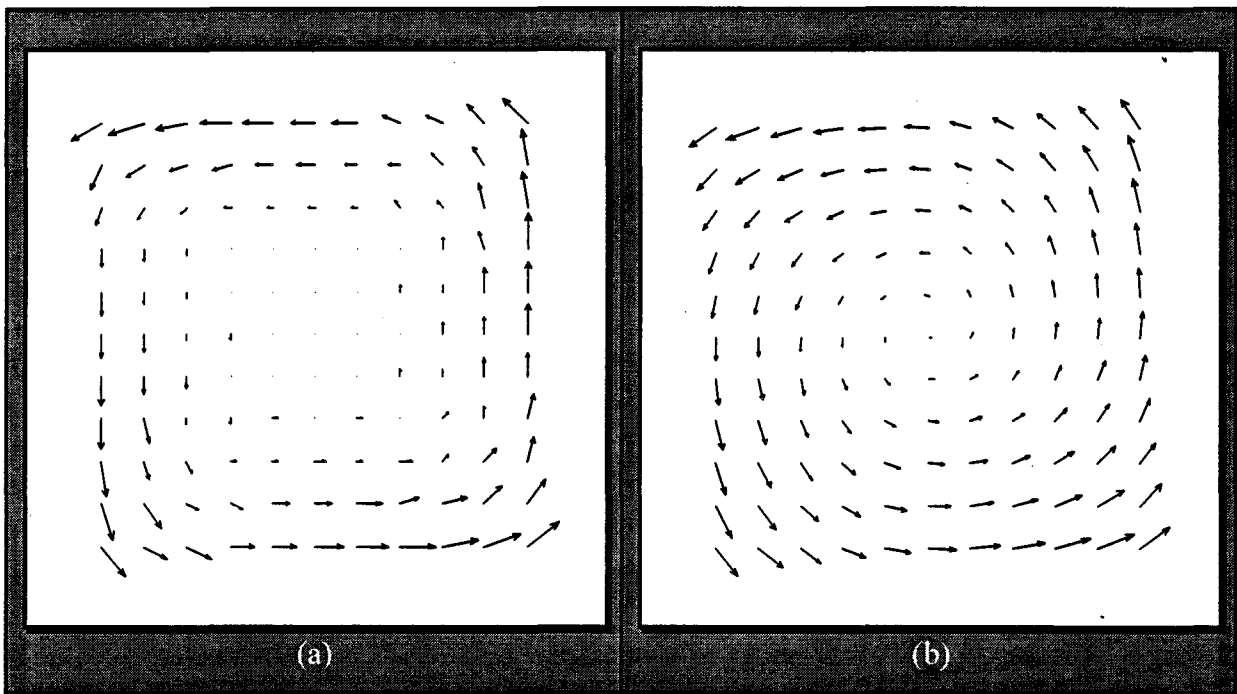


Figure 13 : champ de vitesses rotationnel par le réseau avec (a) $N=20$ et (b) $N=200$ neurones.

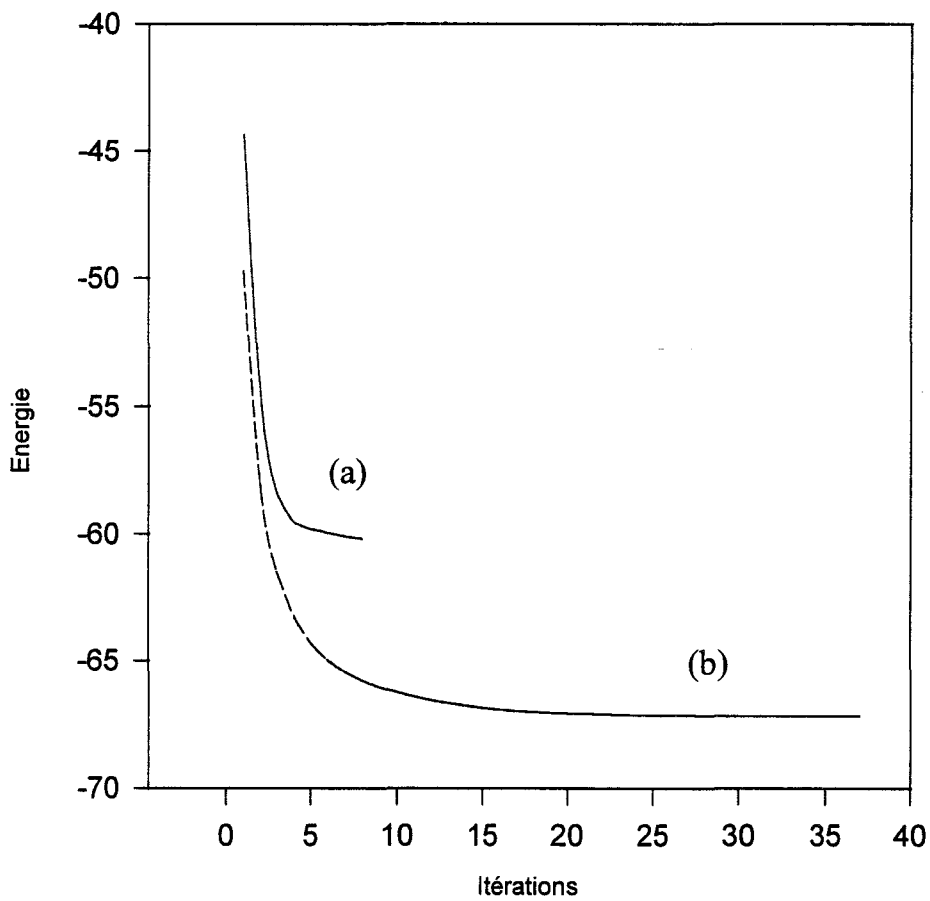


Figure 14 : Evolution de l'énergie du réseau pour (a) $N=20$ et (b) $N=200$

6.3.1.3 cas d'un mouvement divergent

Le réseau est ainsi capable de détecter des mouvements apparents plus complexes comme le champ divergent, Figure 15. Le champ est obtenu pour un nombre de neurones N égal à 100 et $\alpha=100$. Il est le résultat d'un mouvement axial, c'est à dire le long de l'axe optique de la caméra.

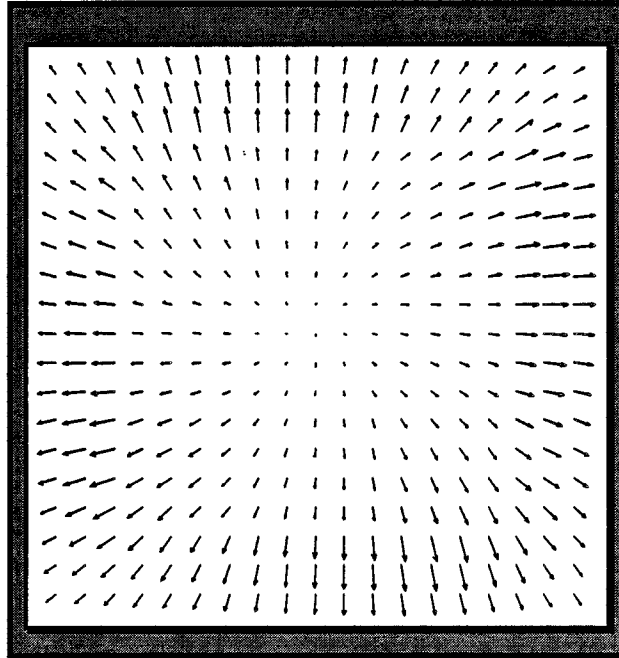


Figure 15 : champ divergent

6.3.1.4 Paramètre de quantification et minimum d'énergie

Afin de rechercher la solution optimale, nous avons effectué une série d'expérimentation sur l'évolution du réseau lorsque le paramètre de quantification k varie. Ces tests ont été réalisés sur le carré en translation de la Séquence 1. Nous avons souligné précédemment que le produit $k \cdot N$ traduisait la dynamique des vecteurs déplacement. Nous avons ainsi réalisé une série de mesures de l'énergie finale du réseau pour différentes configurations du nombre de neurones ($N=10$, $N=20$, $N=100$). Pour chaque cas de figure, les valeurs du paramètre de quantification s'échelonnent de 0.01 à 0.2. La Figure 16 présente les courbes donnant les valeurs de l'énergie finale de chaque réseau en fonction de k pour N fixé. Nous constatons à la vue de ces courbes qu'elles présentent toutes un minimum pour une certaine valeur de pas de quantification, que l'on dénommera k_{optimal} . Il est donc possible d'asservir automatiquement le calcul du paramètre de quantification k en se basant sur la recherche du minimum de ces courbes. Toutefois, il apparaît que les trois minima locaux

correspondent à une même dynamique, autrement dit le produit $k \cdot N$ est identique. Dans l'exemple actuel, il est approximativement égal à 0,875.

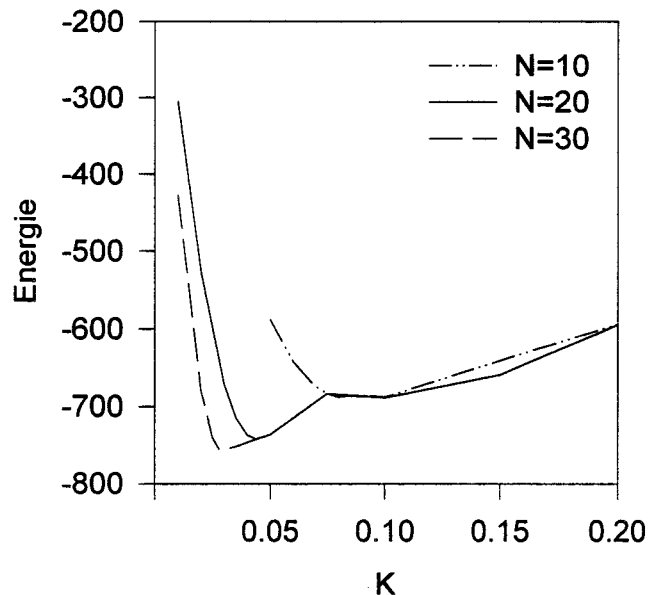


Figure 16 : Courbes d'énergie en fonction du paramètre de quantification k pour $N=10$, $N=20$ et $N=30$

Ce résultat permet de déduire que l'estimation du mouvement obtenue par notre approche ne dépend pas de la configuration (k, N) choisie mais uniquement de la séquence traitée.

6.3.1.5 Cas d'images bruitées

Afin de compléter cette analyse, nous nous sommes intéressés à la capacité du réseau dans le cas d'images de synthèse bruitées. Nous conservons la Séquence 1 comme support de test auquel on ajoute un bruit de distribution normale, pour différentes valeurs de rapport signal sur bruit. Cette série d'expérimentation se fait pour des valeurs de variances $\sigma=6$, $\sigma=10$, $\sigma=20$, $\sigma=40$. L'évaluation des solutions se fait en terme d'erreurs moyennes d'amplitude et de direction des vecteurs vitesses sur le principe de la relation (6.29). La valeur du pas de quantification est évaluée pour chaque configuration du réseau, $N=20$ et $N=100$, en fonction de la "dynamique optimale" déterminée précédemment et égale à 0,875. Le Tableau 3 expose les résultats obtenus pour les différentes valeurs de variance en les comparant à ceux obtenus

par la méthode de Horn & Schunk. Ces mêmes caractéristiques sont fournies pour le cas de l'image sans bruit.

<i>Erreur</i> <i>Variance</i>	<i>Horn & Schunk</i>		<i>Réseau de neurones</i> <i>N=20</i>		<i>Réseau de neurones</i> <i>N=100</i>	
	Amplitude (%)	Direction (°)	Amplitude (%)	Direction (°)	Amplitude (%)	Direction (°)
$\sigma=0$	14.79	4.4381	35.0264	12.4727	17.7603	6.0243
$\sigma=6$	15.0097	4.6211	29.7207	10.4975	16.4321	5.6407
$\sigma=10$	17.271	4.9713	32.0053	11.5399	17.089	5.7649
$\sigma=20$	25.3246	7.4466	35.2118	12.0277	22.95	6.7928
$\sigma=40$	39.5432	12.0515	38.8353	13.5560	30.5319	9.2109

Tableau 3 : Erreurs moyennes en amplitude et direction

Une première constatation est que les données obtenues par le réseau sont proches de celles réalisées par la méthode de Horn & Schunk. De même que dans le cas d'une image sans bruit, les erreurs moyennes sont d'autant plus faibles que le nombre de neurones augmente. Il existe pratiquement un coefficient de 2 entre le réseau fonctionnant avec 20 neurones et celui avec $N=100$. Une seconde constatation est que, si on compare la méthode continue de Horn & Schunk aux résultats obtenus par notre réseau, la solution est meilleure lorsque le bruit augmente (exemple si $\sigma=40$, avec un nombre de neurones $N=100$, $\varepsilon_{amp}=30.5319$ et $\varepsilon_{ang}=9.2109$ pour Horn & Schunk, $\varepsilon_{amp}=39.5432$ et $\varepsilon_{ang}=12.0515$).

Du point de vue de la carte des champs de vitesse, la Figure 17 montre un champ plus régulier dans le cas de notre approche, Figure 17.a, comparé à celui obtenu par la méthode de Horn & Schunk, Figure 17.b, pour la séquence d'images avec un bruit additif de variance $\sigma=40$.

Nous pouvons conclure sur ces remarques que lorsque le bruit σ au sein de l'image n'est pas prépondérant vis à vis du bruit de quantification au sein du réseau, les différences existantes entre le réseau et la méthode continue de Horn & Schunk sont dues au phénomène de quantification. Au contraire, quand σ augmente ($\sigma \approx 40$ pour $N=20$ et $\sigma \approx 15$ pour $N=100$), le bruit de quantification devient inférieur au bruit de l'image et le champ de vitesses final obtenu par le réseau est plus robuste.

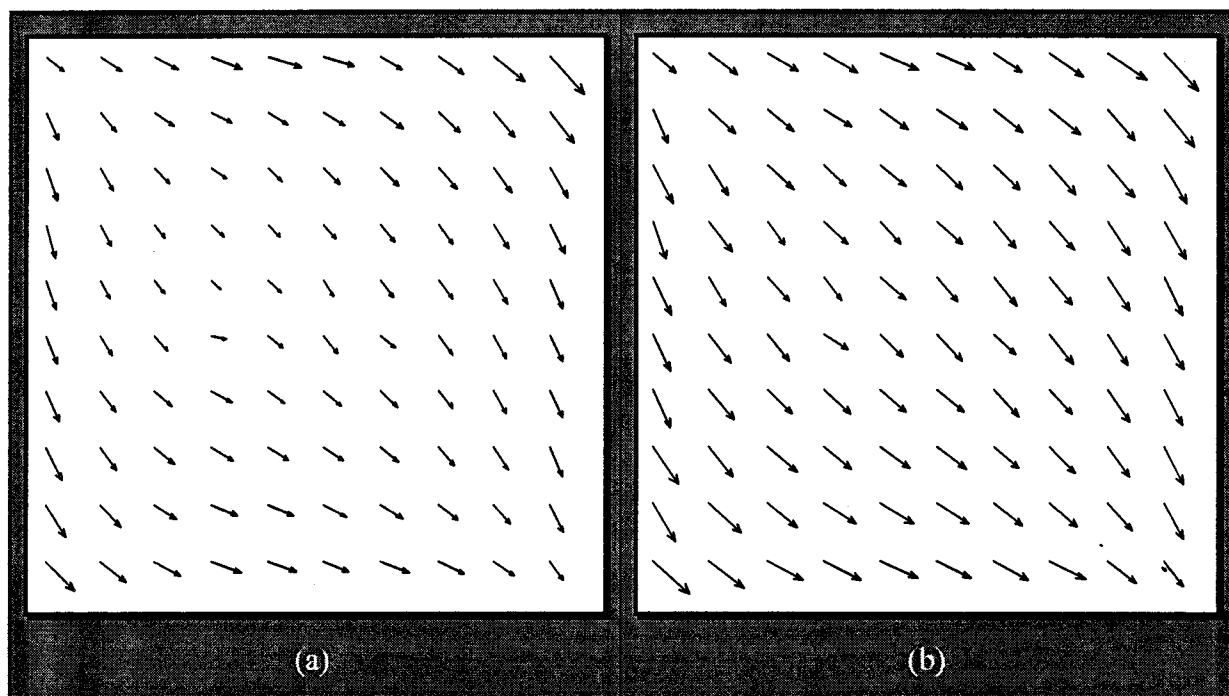


Figure 17 : Champs de vitesse obtenus sur la séquence d'images avec un bruit de variance $\sigma=40$ par
(a) la méthode de Horn & Schunk
(b) le réseau avec 100 neurones

6.3.2 Séquences d'images réelles

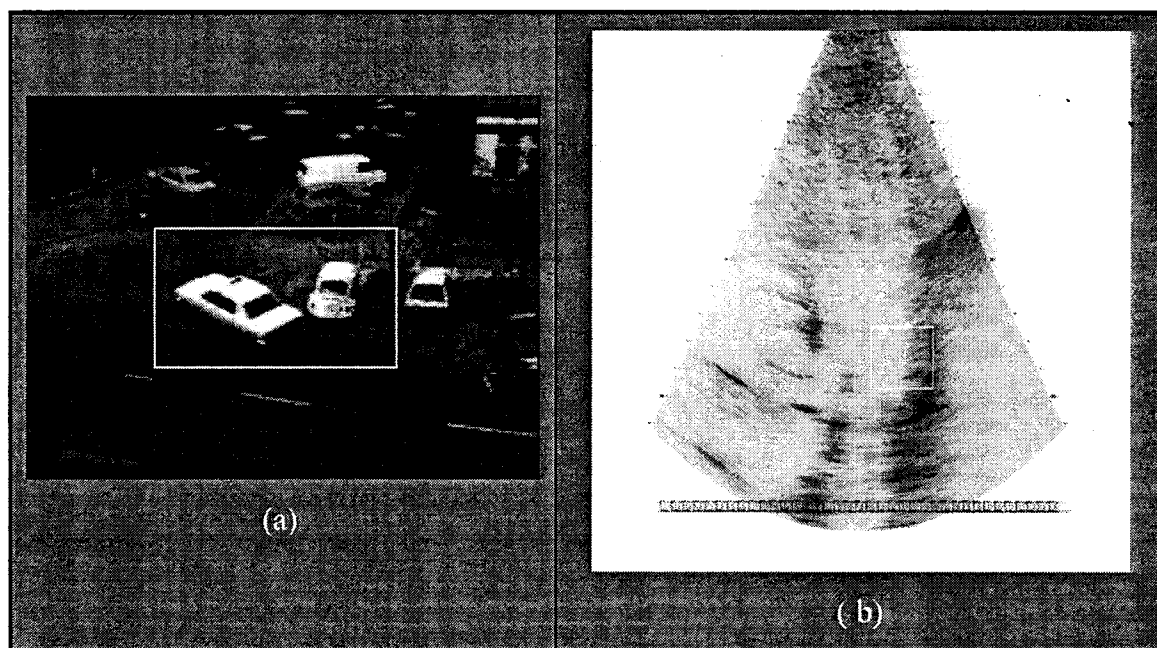
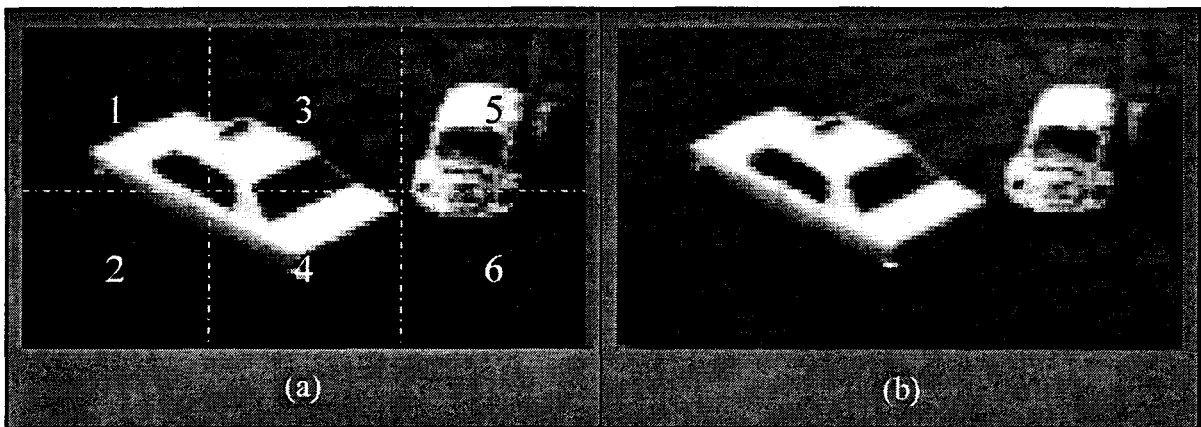


Figure 18 : Image 1 de la séquence
(a) d'une scène de rue à Hambourg
(b) échocardiographique du ventricule gauche (image inversée)

Nous avons testé notre réseau de neurones sur deux types de séquences d'images réelles différentes. La Figure 18 présente les premières images de chacune de ces deux séquences. La première d'entre elle, Figure 18.a est une scène classique de rue de l'université de Hambourg où quatre objets sont en mouvement, le taxi, la voiture dans le coin en bas à gauche, le van dans le coin en bas à droite, et un piéton. La seconde figure est issue d'une séquence d'images échocardiographiques du ventricule gauche. Une fenêtre, caractérisée par un rectangle blanc, sur une zone d'intérêt a été définie sur chacune de ces deux séquences. La Séquence 3 illustre cette fenêtre localisée autour du taxi blanc. Respectivement, la Séquence 4 montre la zone sélectionnée au sein de laquelle l'estimation sera effectuée.



Séquence 3 : zone autour du taxi sélectionné

La Figure 19 représente les champs des vitesses obtenus respectivement par la méthode classique de Horn & Schunk, Figure 19.a, et par le réseau de neurones avec $N=20$ et un pas de quantification k égal à 0.025, Figure 19.b. Dans les deux cas, le terme de lissage α est fixé à 10. Ces résultats ont été acquis après 232 itérations pour Horn & Schunk et après 25 itérations pour le réseau.

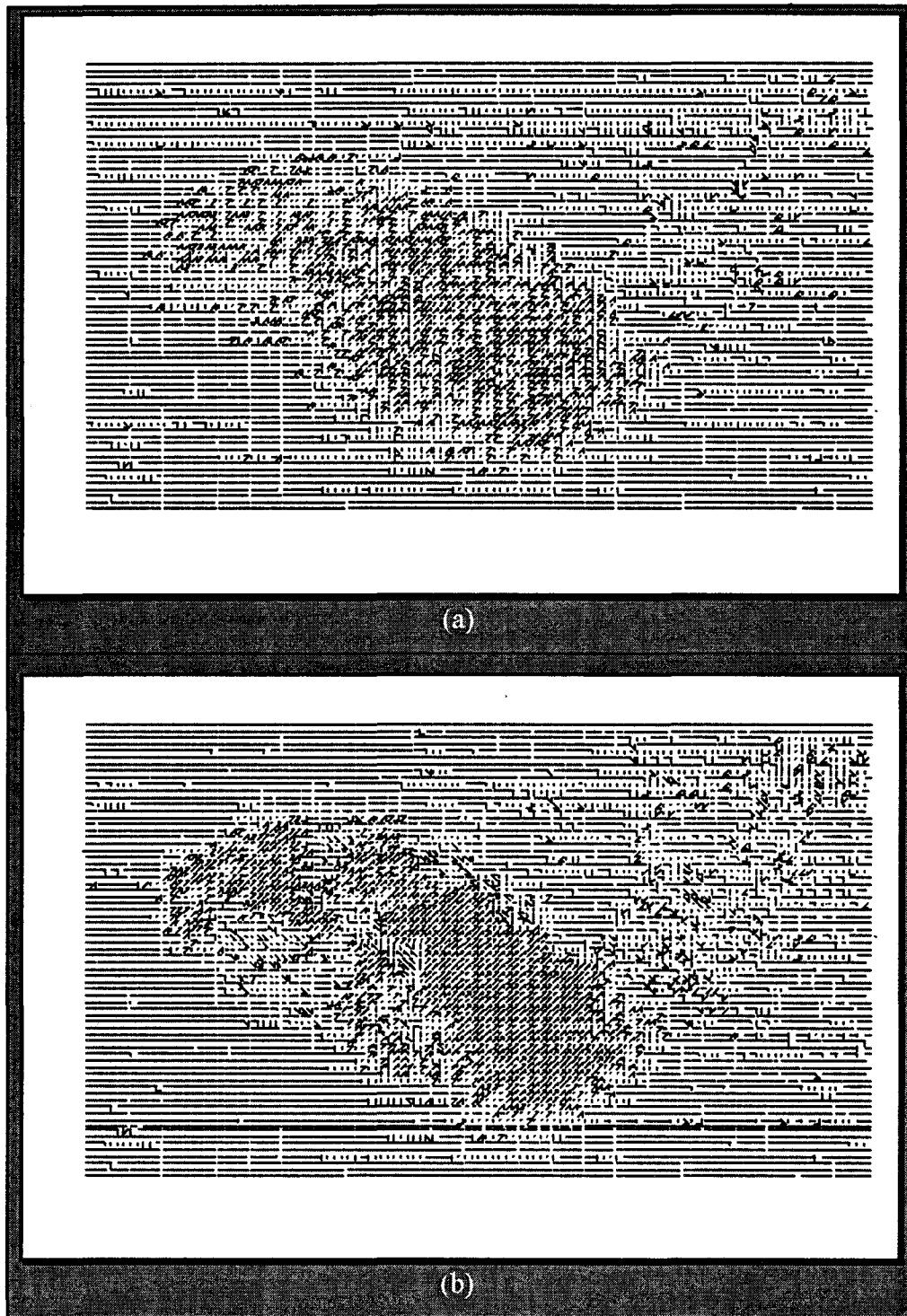


Figure 19 : champs des vecteurs vitesses obtenus pour la pour :
(a) méthode de Horn & Schunk
(b) réseau de neurones avec $N=20$

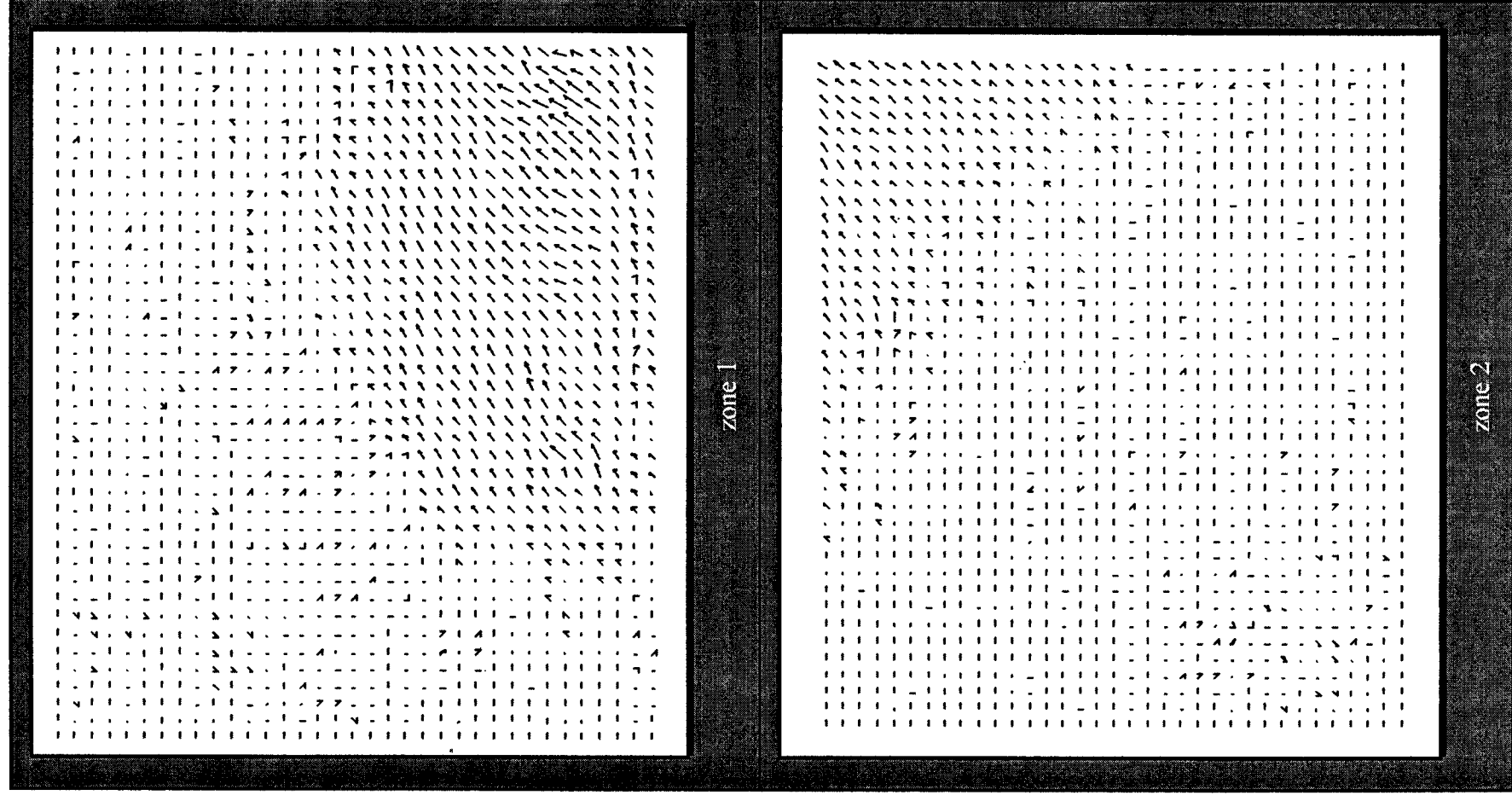


Figure 20.a : Méthode de Horn & Schunk : zones 1 et 2

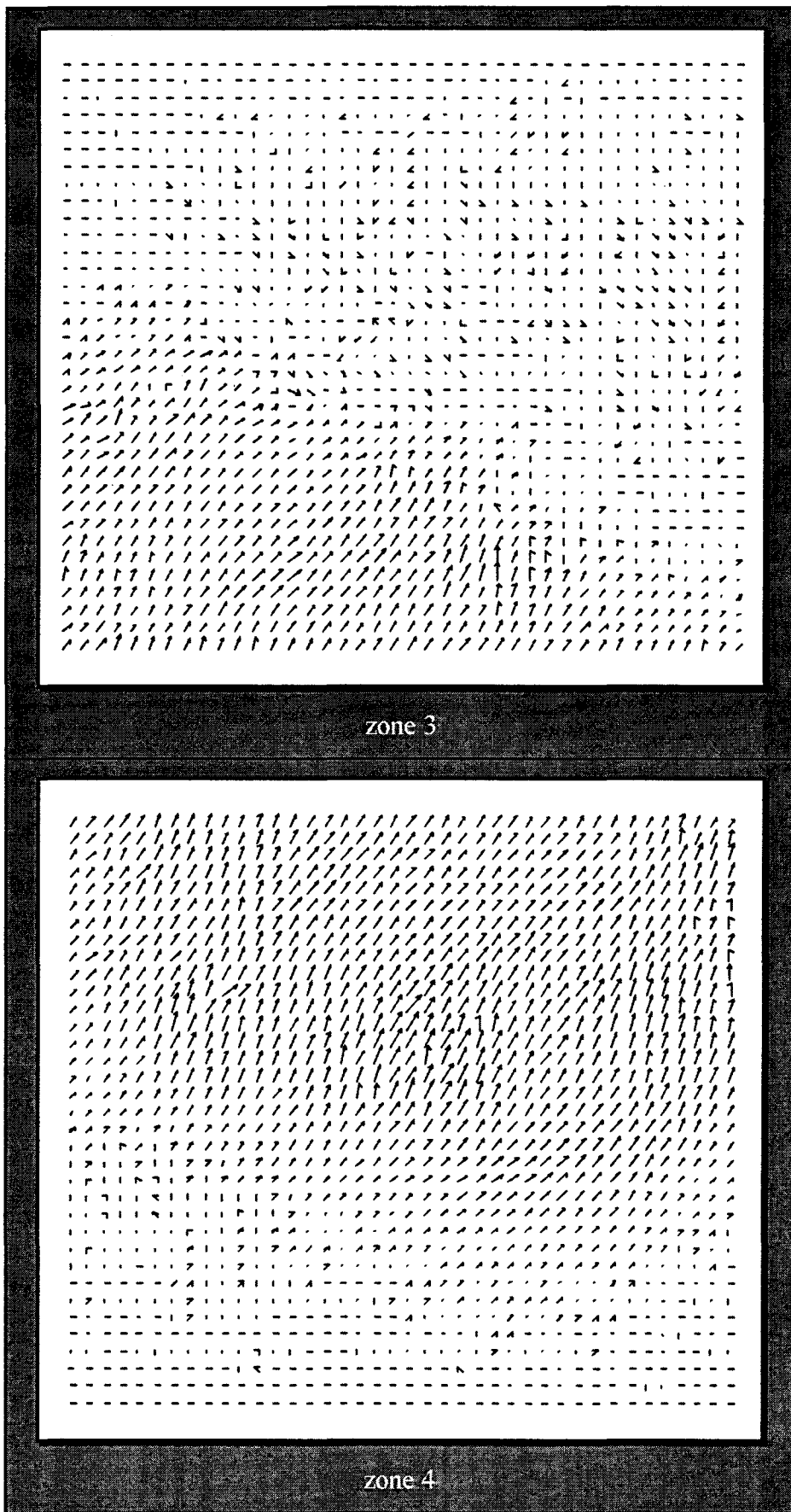


Figure 20.b : Méthode de Horn & Schunk : zones 3 et 4

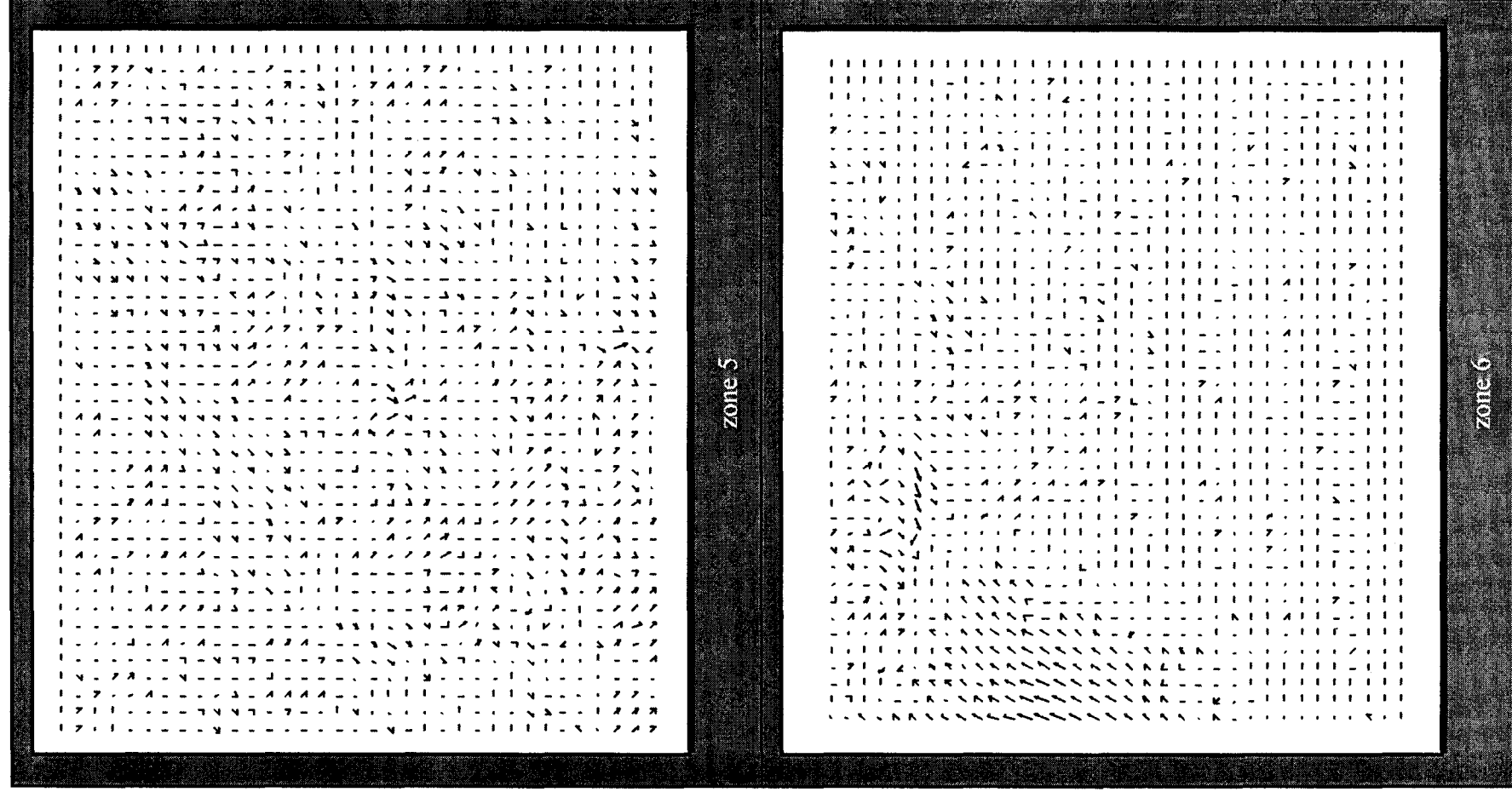


Figure 20.c : Méthode de Horn & Schunk : zones 5 et 6

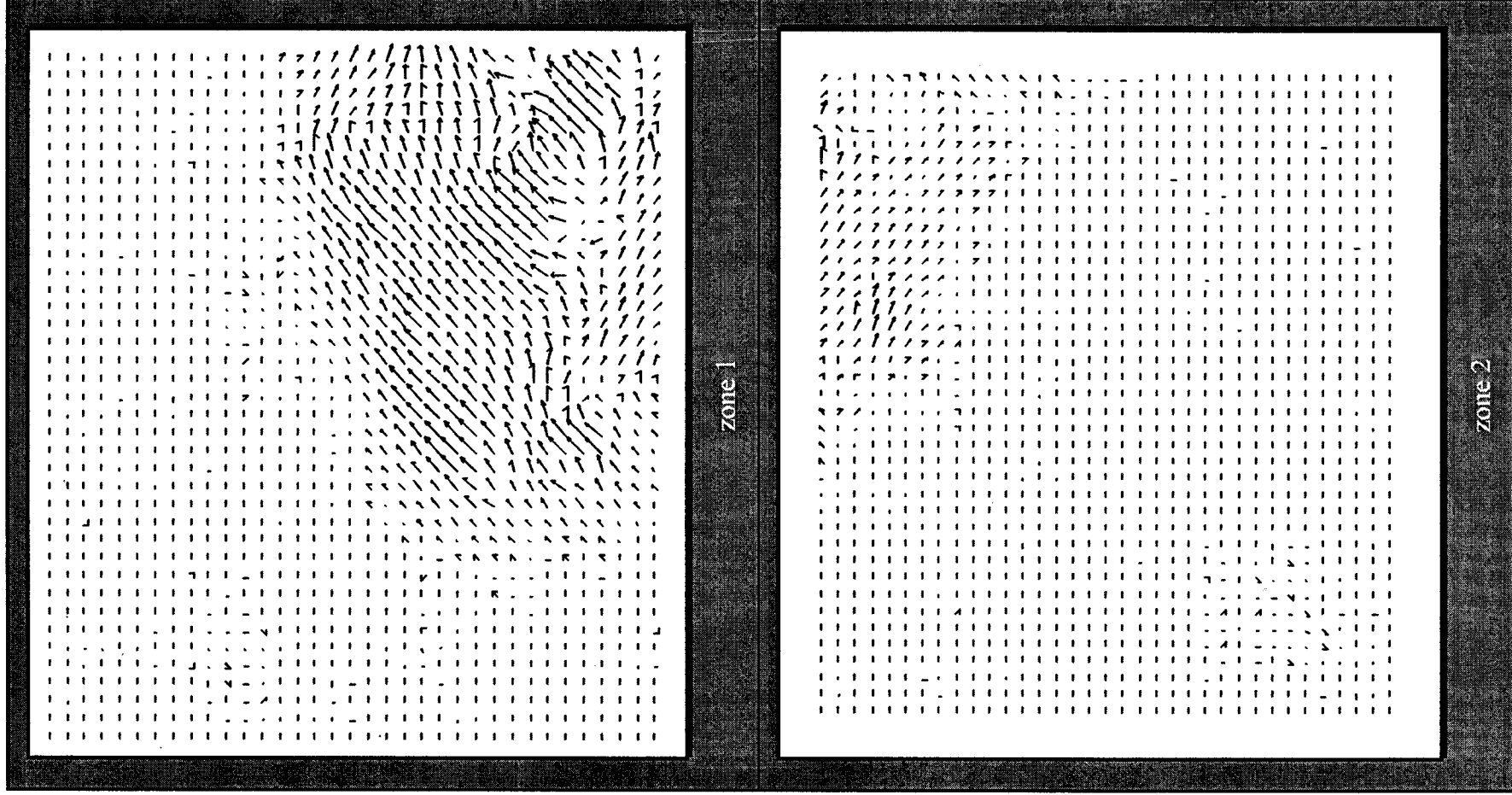


Figure 21.a : Réseau de neurones : zones 1 et 2

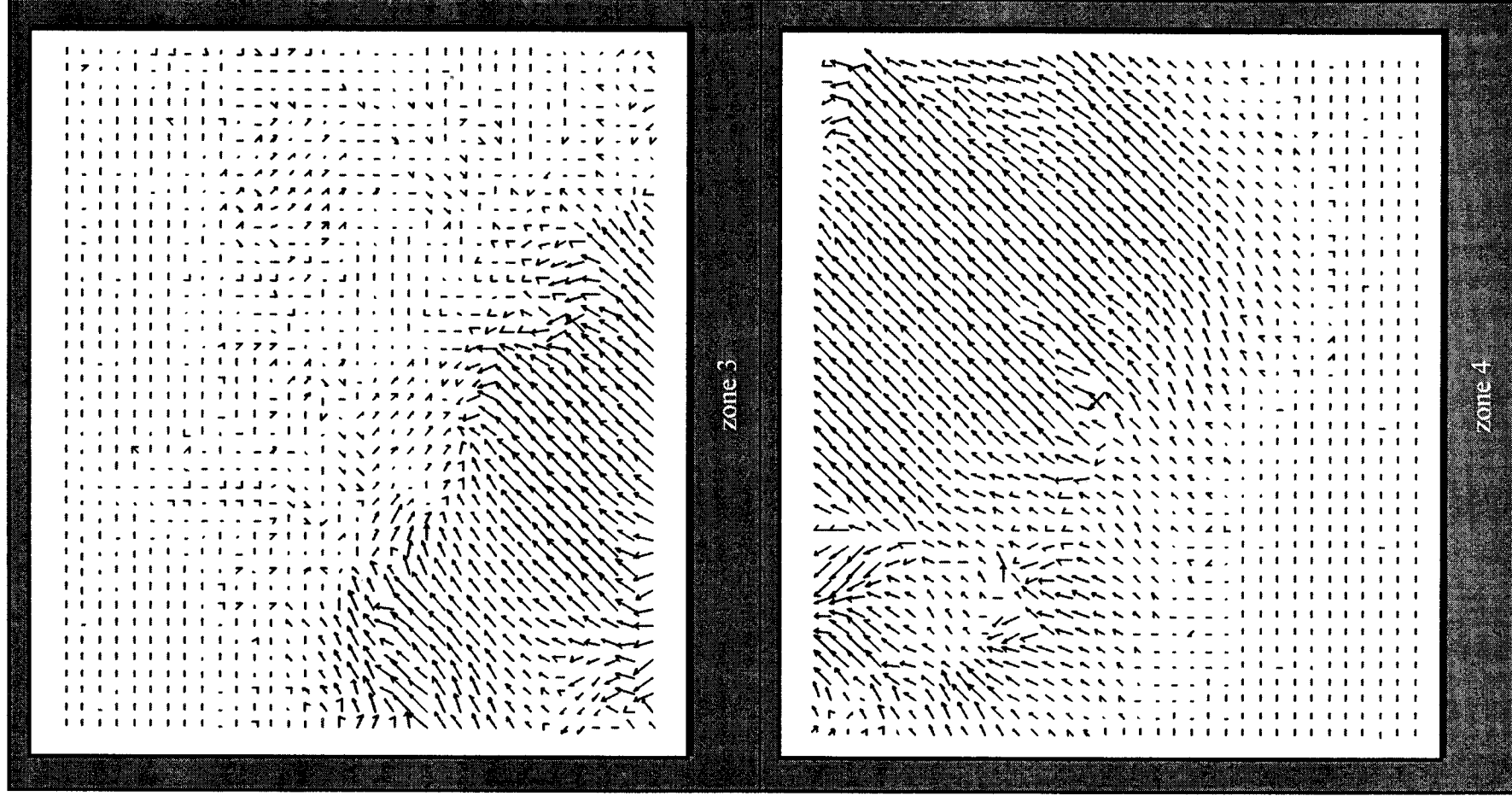
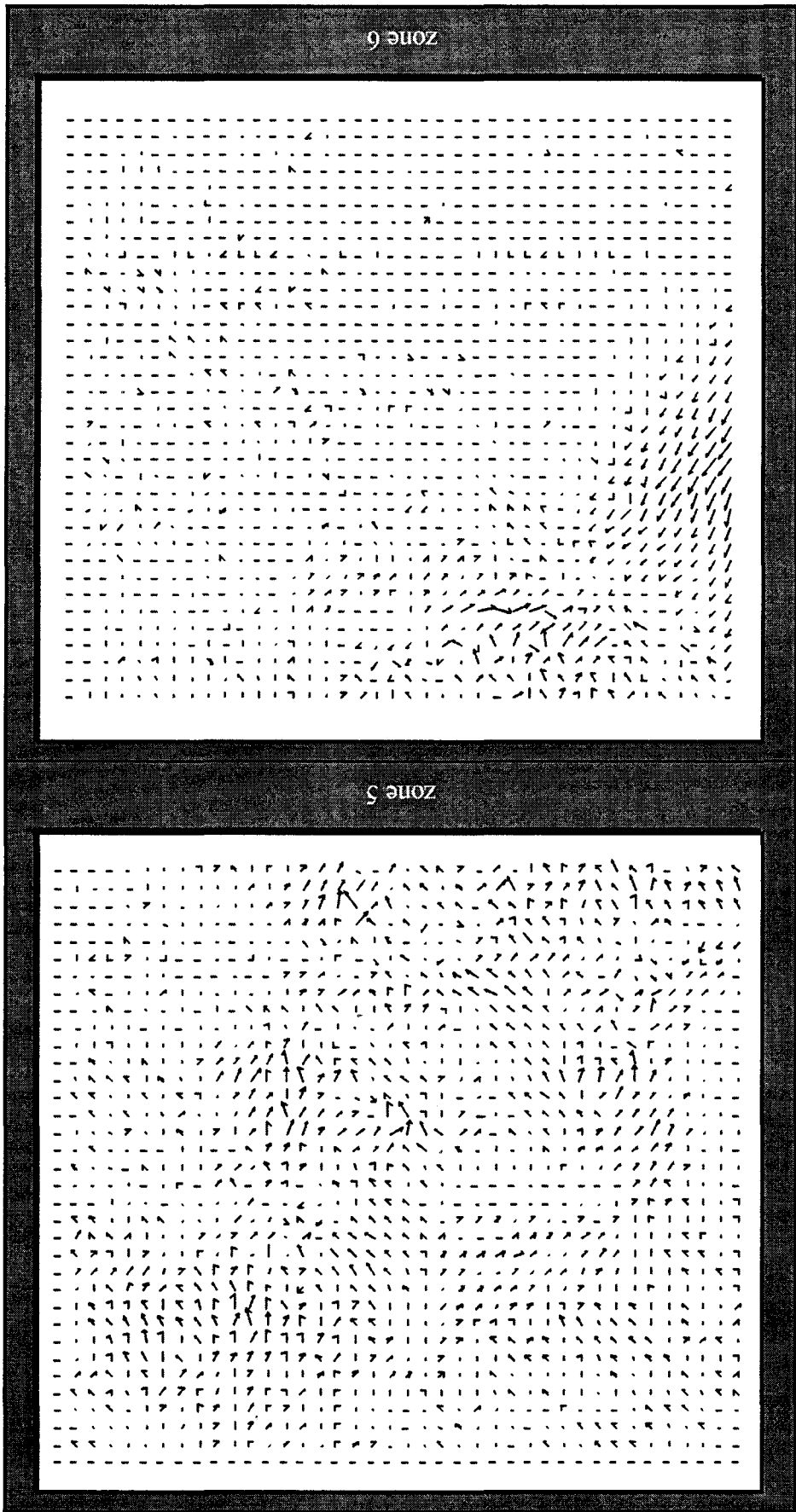
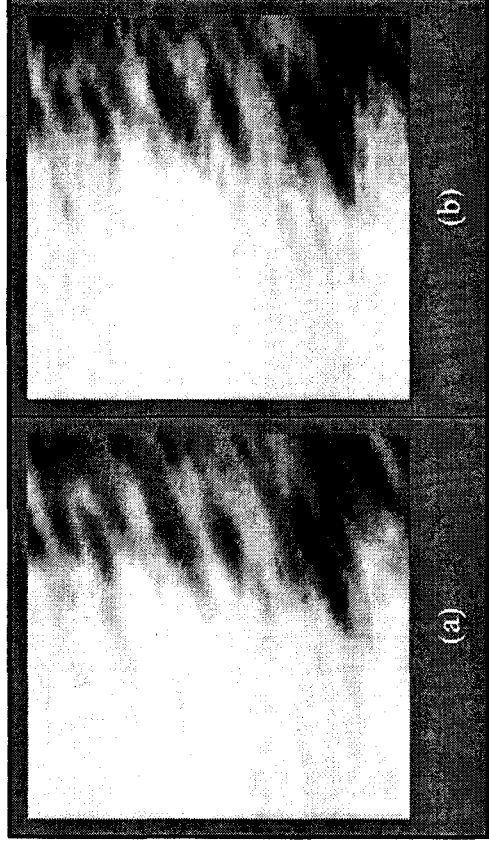


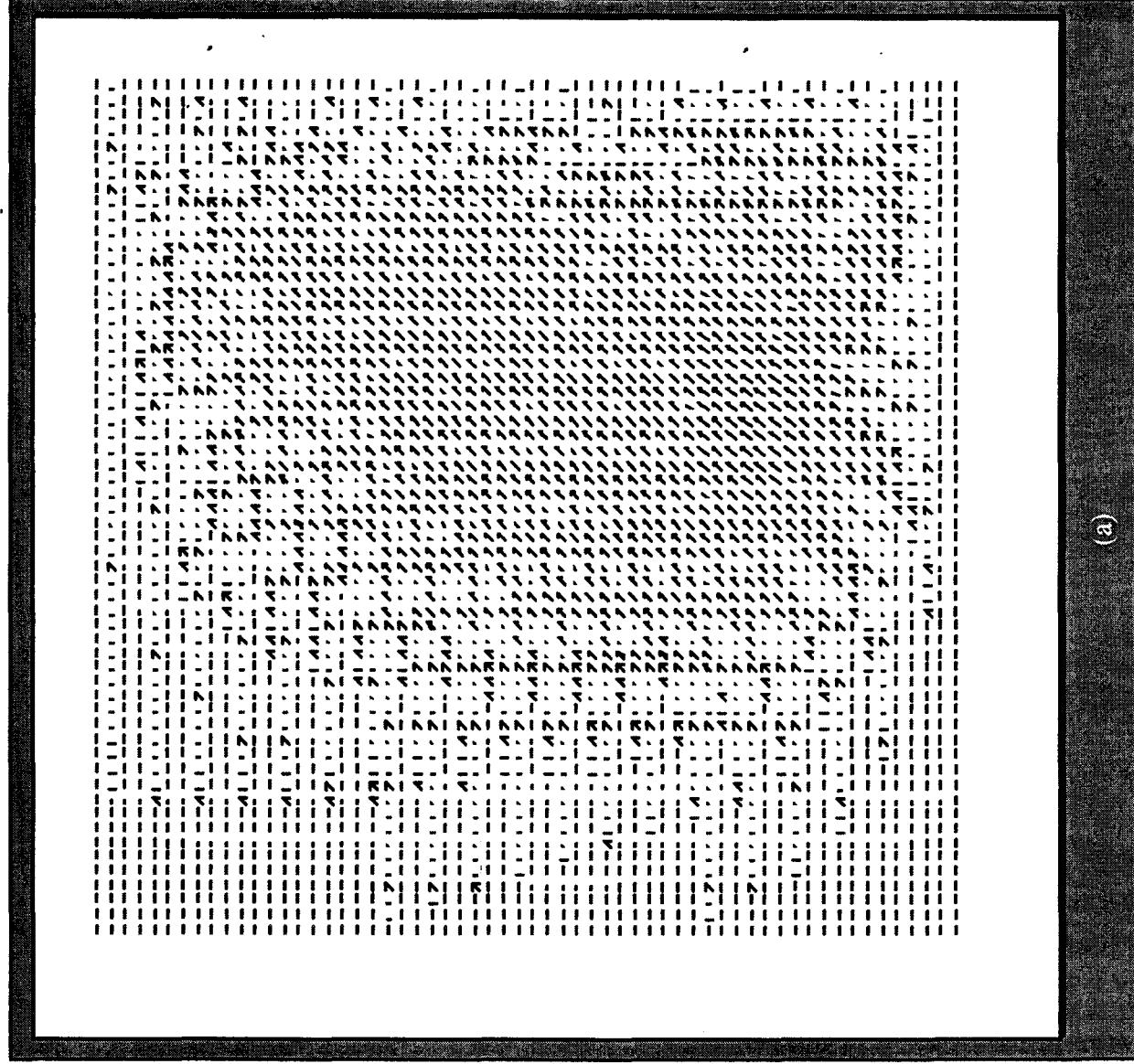
Figure 21.b : Réseau de neurones : zones 3 et 4

Figure 21.c : Réseau de neurones : zones 5 et 6





Séquence 4 : zone de la paroi du ventricule gauche étudiée



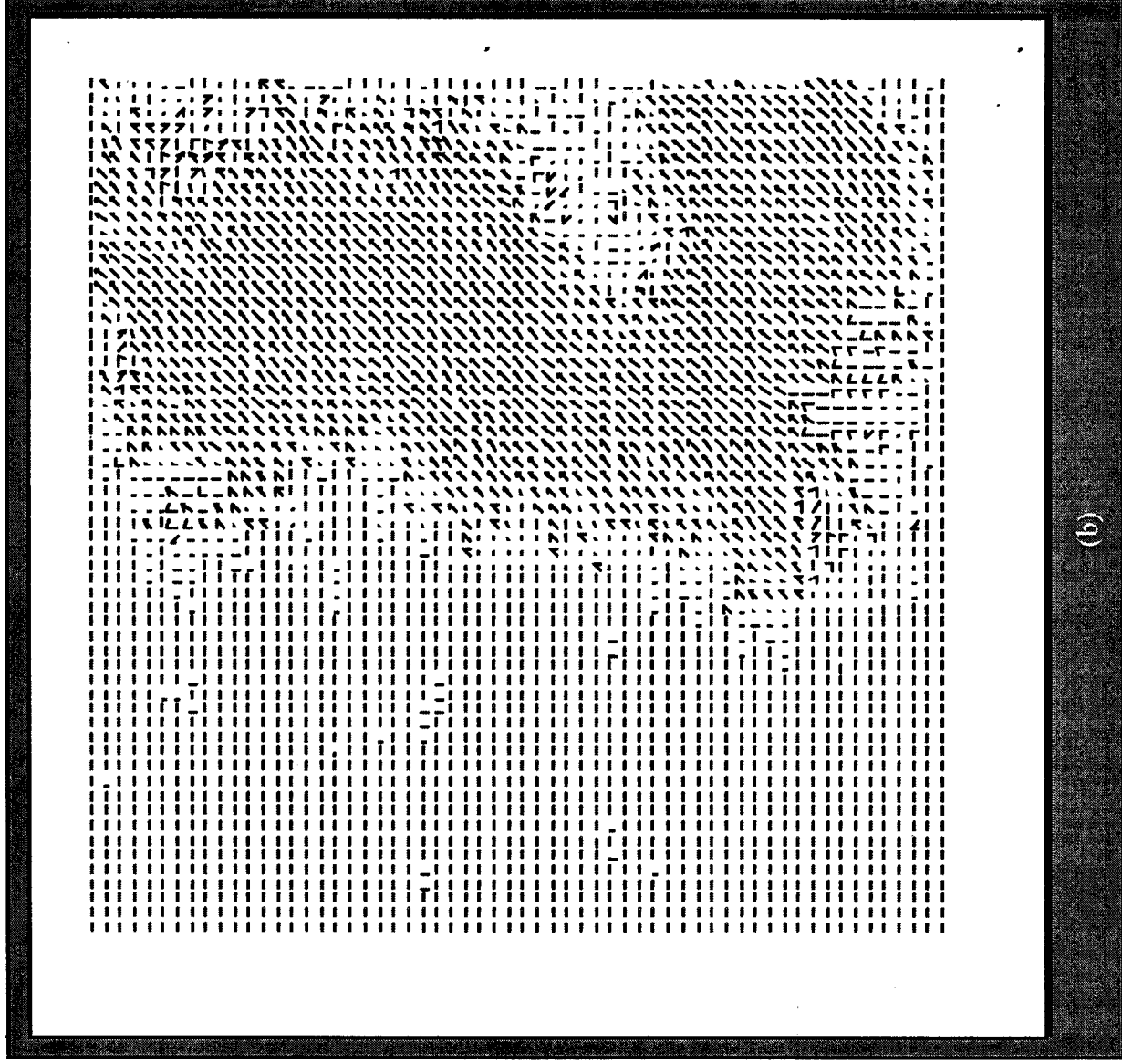


Figure 22 : champs des vecteurs vitesses obtenus pour la pour :

(a) méthode de Horn & Schunk

(b) réseau de neurones avec $N=50$

Les résultats de l'estimation du mouvement apparent sur l'autre séquence sont présentées Figure 22. Le terme de régularisation α est fixé ici à 100. Le réseau de neurones est constitué de 50 neurones avec une valeur de 0.005 pour le paramètre de quantification k . Nous constatons sur cet exemple que les zones en mouvement correspondent aux zones texturées de l'image [Yacoubi, Lelaurin, Devlaminck, & Dubus, 1995b].

De plus, si l'on compare la carte des champs obtenue par l'approche de Horn & Schunk à celle de notre réseau, cette zone en mouvement est mieux localisée. Ceci est dû à la quantification qui permet de tronquer les faibles valeurs propagées par le terme de lissage.

6.4 Extension à des mouvements d'objets déformables

Le réseau est expérimenté à titre de comparaison en utilisant une fonctionnelle dédiée aux mouvements d'objets déformables. L'approche du flot optique sous l'aspect préalablement proposé impose certaines limites sur les types de mouvement, en particulier le fait que ces derniers soient rigides. Autrement dit, l'objet n'a subi aucune déformation, par hypothèse, entre l'instant t et l'instant $t+1$. Cependant, dans de nombreuses applications (l'imagerie médicale [Nastar, 1994] [Pun, Gerig, & Ratib, 1994], suivi de nuages en vue de prévisions météorologiques [Aggarwal & Duda, 1975], mesure du champ des matériaux sous charge mécanique [Yang, 1992], ...), cette approche n'est plus adéquate et il est nécessaire d'avoir recours à des contraintes sur la solution finale qui prennent en compte les déformations possibles.

Nous conservons l'équation de contrainte (6.6) auquel nous rajoutons une équation caractérisant la conservation de la quantité de mouvement. Cette équation donne une nouvelle contrainte de régularisation dont l'expression est la suivante dans le cas de déformations de type élastiques [Devlaminck & Dubus, 1996] :

$$\alpha_1^2 \cdot \left[\frac{\partial w_x}{\partial x} + \frac{\partial w_y}{\partial y} \right]^2 + \alpha_2^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 + \frac{1}{2} \cdot \left(\frac{\partial w_x}{\partial y} + \frac{\partial w_y}{\partial x} \right)^2 \right] \quad (6.27)$$

La fonctionnelle à minimiser a alors pour nouvelle expression :

$$E = \iint \gamma^2 \cdot \left\{ \left(f_x \cdot w_x + f_y \cdot w_y + f_t \right)^2 + \alpha_1^2 \cdot \left[\frac{\partial w_x}{\partial x} + \frac{\partial w_y}{\partial y} \right]^2 + \alpha_2^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 + \frac{1}{2} \cdot \left(\frac{\partial w_x}{\partial y} + \frac{\partial w_y}{\partial x} \right)^2 \right] \right\} \cdot dx \cdot dy \quad (6.28)$$

Les deux coefficients α_1 et α_2 sont donnés par les relations :

$$\alpha_1 = \frac{Y}{2 \cdot (1 + \nu)} \quad \alpha_2 = \frac{Y}{3 \cdot (1 - 2 \cdot \nu)} \quad (6.29)$$

où Y représente le module de Young et ν le coefficient de Poisson.

6.4.1 Nouveaux paramètres du réseau

Après la phase d'identification sur le modèle décrit au chapitre 5, les nouvelles valeurs des poids synaptiques sont les suivantes [Devlaminck, Lelaurin, Yacoubi, & Dubus, 1996] :

➤ **Poids synaptiques**

- ◆ Pour les sous-réseaux $w_{(i,j)x}^+$ et $w_{(i,j)x}^-$
- Connexions intra sous-réseaux

$$T_{i,j,h^+;m,n,l^+} = -2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)x}^2 + \frac{2 \cdot \alpha_1^2 + 3 \cdot \alpha_2^2}{2} \right) \cdot \delta_{(i,j),(m,n)} - \left(\frac{2 \cdot \alpha_1^2 + 3 \cdot \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_1} \right. \\ \left. + \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{4} \right) \cdot \delta_{(i,j),(m,n) \oplus s_2} - \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{4} \right) \cdot \delta_{(i,j),(m,n) \oplus s_3} \right] \quad (6.30a)$$

$$T_{i,j,h^-;m,n,l^-} = T_{i,j,h^+;m,n,l^+}$$

- Connexions inter sous-réseaux $w_{(i,j)x}^+ \rightarrow w_{(i,j)x}^-$

$$T_{i,j,h^+;m,n,h^-} = T_{i,j,h^-;m,n,h^+} = -T_{i,j,h^+;m,n,l^+} \quad (6.30b)$$

- Connexions inter composantes $w_{(i,j)x} \rightarrow w_{(i,j)y}$

- * $w_{(i,j)x}^+ \rightarrow w_{(i,j)y}^+$ et $w_{(i,j)x}^- \rightarrow w_{(i,j)y}^-$

$$T_{i,j,h^+;m,n,p^+} = -\gamma^2 \cdot k^2 \cdot \left[f_{(i,j)x} \cdot f_{(i,j)y} \cdot \delta_{(i,j),(m,n)} \right. \\ \left. - \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_4} + \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_5} \right] \quad (6.30c)$$

$$T_{i,j,h^-;m,n,p^-} = T_{i,j,h^+;m,n,p^+}$$

- * $w_{(i,j)x}^+ \rightarrow w_{(i,j)y}^-$ et $w_{(i,j)x}^- \rightarrow w_{(i,j)y}^+$

$$T_{i,j,h^+;m,n,p^-} = T_{i,j,h^-;m,n,p^+} = -T_{i,j,h^+;m,n,p^+} \quad (6.30d)$$

- ◆ Pour les sous-réseaux $w_{(i,j)y}^+$ et $w_{(i,j)y}^-$

- Connexions intra sous-réseaux

$$\begin{aligned} T_{i,j,p^+;m,n,q^+} = -2 \cdot \gamma^2 \cdot k^2 \cdot \left[\left(f_{(i,j)y}^2 + \frac{2 \cdot \alpha_1^2 + 3 \cdot \alpha_2^2}{2} \right) \cdot \delta_{(i,j),(m,n)} - \left(\frac{2 \cdot \alpha_1^2 + 3 \cdot \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_1} \right. \\ \left. - \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{4} \right) \cdot \delta_{(i,j),(m,n) \oplus s_2} + \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{4} \right) \cdot \delta_{(i,j),(m,n) \oplus s_3} \right] \end{aligned} \quad (6.31a)$$

$$T_{i,j,p^-;m,n,q^-} = T_{i,j,p^+;m,n,q^+}$$

- Connexions inter sous-réseaux $w_{(i,j)y}^+ \begin{matrix} \rightarrow \\ \leftarrow \end{matrix} w_{(i,j)y}^-$

$$T_{i,j,p^+;m,n,p^-} = T_{i,j,p^-;m,n,p^+} = -T_{i,j,p^+;m,n,q^+} \quad (6.31b)$$

- Connexions inter composantes $w_{(i,j)y} \rightarrow w_{(i,j)x}$

- * $w_{(i,j)y}^+ \rightarrow w_{(i,j)x}^+$ et $w_{(i,j)y}^- \rightarrow w_{(i,j)x}^-$

$$\begin{aligned} T_{i,j,p^+;m,n,h^+} = -\gamma^2 \cdot k^2 \cdot \left[f_{(i,j)y} \cdot f_{(i,j)x} \cdot \delta_{(i,j),(m,n)} \right. \\ \left. - \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_4} + \left(\frac{2 \cdot \alpha_1^2 + \alpha_2^2}{8} \right) \cdot \delta_{(i,j),(m,n) \oplus s_5} \right] \end{aligned} \quad (6.31c)$$

$$T_{i,j,p^-;m,n,h^-} = T_{i,j,p^+;m,n,h^+}$$

- * $w_{(i,j)y}^- \rightarrow w_{(i,j)x}^+$ et $w_{(i,j)y}^+ \rightarrow w_{(i,j)x}^-$

$$T_{i,j,p^+;m,n,h^-} = T_{i,j,p^-;m,n,h^+} = -T_{i,j,p^+;m,n,h^+} \quad (6.31d)$$

où S_1, S_2, S_3, S_4, S_5 représentent une série d'index avec :

$$S_1 = \{(-1, -1), (-1, 1), (1, -1), (1, 1)\}, S_2 = \{(-1, 0), (1, 0)\} \quad (6.32)$$

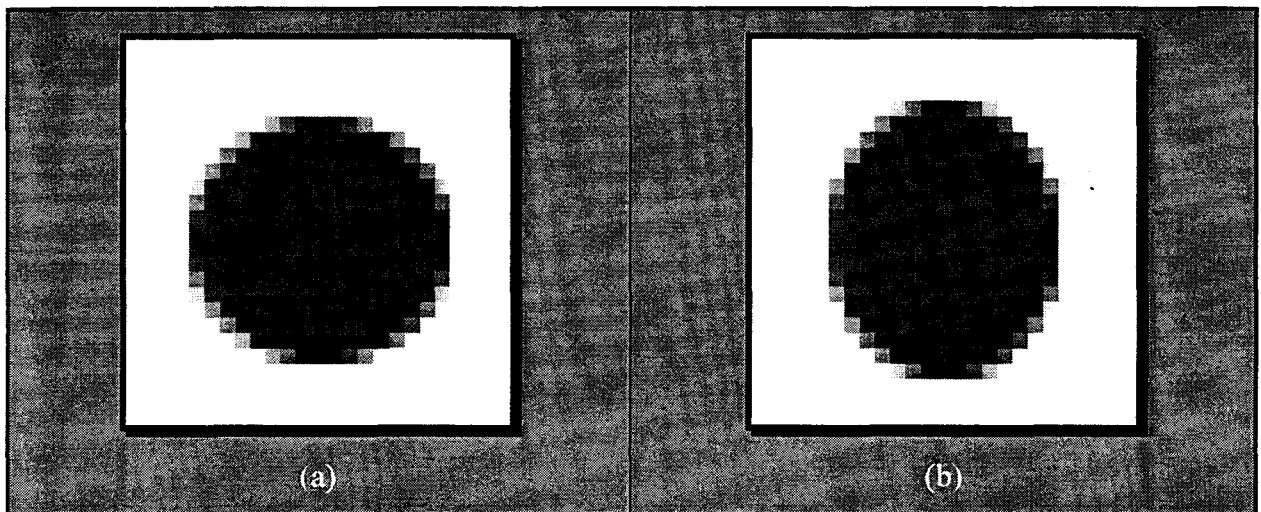
$$S_3 = \{(0, -1), (0, 1)\}, S_4 = \{(-1, -1), (1, 1)\}, S_5 = \{(-1, 1), (1, -1)\}$$

➤ *entrées externes*

Les valeurs des entrées externes de ce réseau sont identiques à celles de la fonctionnelle précédente, relations (6.26), puisque la contrainte originelle du problème est conservée et c'est elle qui fournit ces informations au réseau. On vérifie donc bien, comme énoncée au chapitre 5, que la contrainte de régularisation n'intervient que sur la connectique du réseau.

6.4.2 Résultats expérimentaux

Nous présentons ici les résultats obtenus par cette nouvelle architecture neuronale dans le cas de deux séquences d'images de synthèse, Séquence 5 et Séquence 6.



Séquence 5 : Ellipse déformée sans contrainte de compressibilité

Dans cette première séquence, l'ellipse est déformée avec une contrainte d'incompressibilité imposée. Le taux de déformation est de 1.4, suivant les directions horizontale et verticale. Le choix de ν est fixé à 0.49 et correspond à un matériau incompressible. Le réseau se constitue de 100 neurones avec $k=0.007$. Le champ final, Figure 23, est obtenu après 88 itérations.

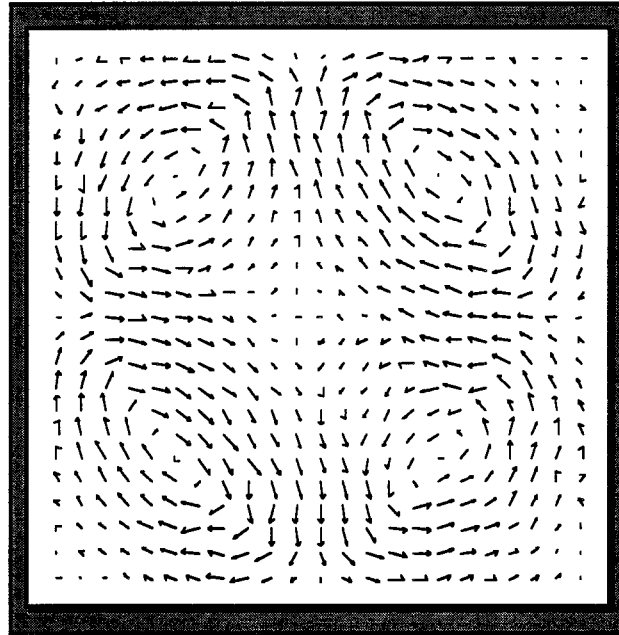
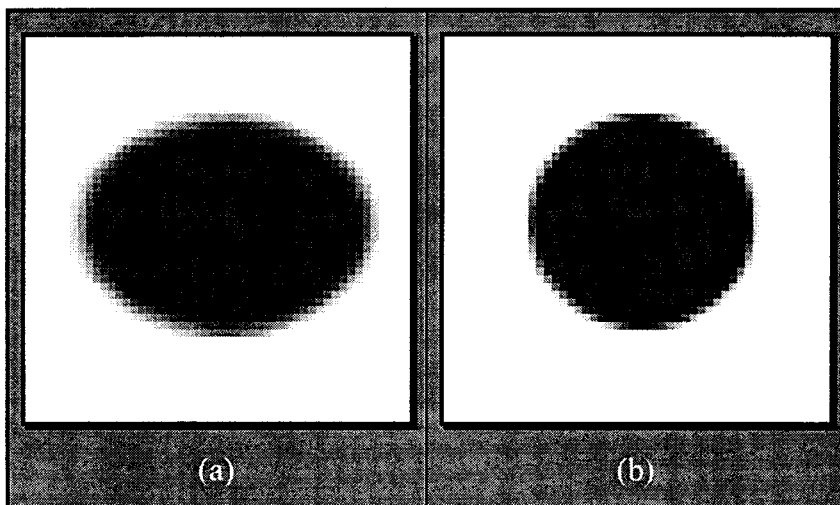


Figure 23 : Champ de vitesse obtenu pour $N=100$

La seconde séquence se constitue d'une ellipse déformée avec un mouvement de compression dissymétrique de 1.4 à gauche et 1.6 à droite. Dans ce cas de figure, il est nécessaire de conserver le terme de divergence sur les vitesses $\text{div}(\vec{w})$ de la relation (6.5). Les paramètres du réseau, entrées externes et poids synaptiques sont donc estimés de nouveau. Chaque sous réseau est toujours constitué, comme dans l'exemple précédent, de 100 neurones avec un pas de quantification k identique égal à 0.01.



Séquence 6 : Ellipse déformée avec un mouvement de compression

Le résultat final, obtenu pour une valeur du coefficient de Poisson ν égale à 0.01, est présenté en Figure 24. Nous pouvons noter que le champ de vecteurs situé sur la partie droite de la figure est plus important que celui situé à gauche. Ceci confirme un taux de compression plus fort à droite qu'à gauche comme énoncé précédemment.

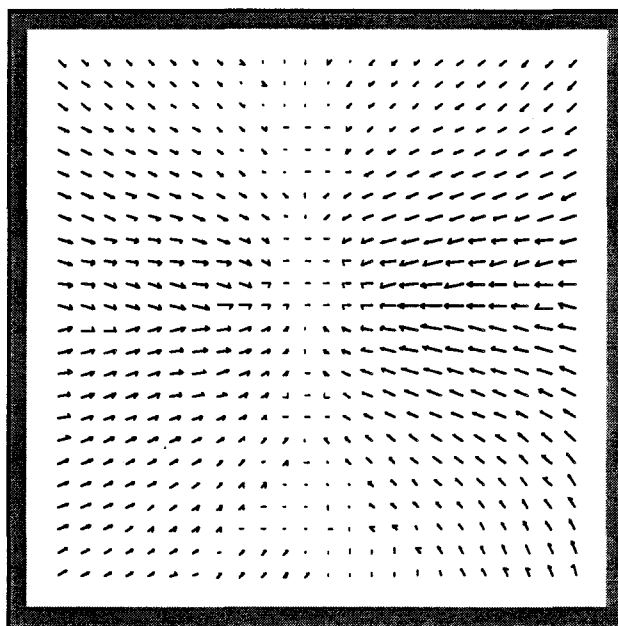


Figure 24 : Champ de vitesse pour $N=100$

6.5 Conclusion

Les résultats observés sur les différentes séquences d'images de synthèse ou réelles ont permis de valider et de montrer certains intérêts de la méthodologie neuronale que nous avons mise en oeuvre dans le cadre de l'estimation du mouvement apparent.

Une première constatation est que la qualité de la solution finale dépend du nombre de neurones affectés à chaque sous-réseau, ainsi que du paramètre de quantification. Une autre constatation réside dans une sensible accélération de la vitesse de convergence du réseau (optimisation des temps de calcul) par son asservissement sur le gradient de sa courbe d'évolution énergétique.

Un autre intérêt concerne le paramètre de quantification k . En effet, il est possible de concevoir une automatisation de la recherche de sa valeur optimale pour une configuration donnée du réseau. De plus, dans le cas d'images bruitées, la quantification de la solution continue permet d'améliorer la solution finale.

Enfin, l'application du réseau au cas de la fonctionnelle destinée au mouvement d'objets déformables montre l'adaptabilité du réseau et le caractère générique de la méthodologie proposée.

Chapitre 7

Conclusion générale

A partir de la constatation que de nombreux problèmes en traitement d'images peuvent être formulés en termes de minimisation d'une fonctionnelle de type quadratique, nous avons été conduit à proposer une structure neuronale capable de s'adapter à un modèle quadratique et de résoudre la fonctionnelle associée. Par sa nature générique, ce réseau et la méthodologie qui l'accompagne permettent de traiter des problèmes classiques en traitement d'images tels que la restauration, la stéréovision, la détection des contours, ou l'estimation du mouvement apparent, puisque tous ces problèmes peuvent s'exprimer sous la forme de la minimisation d'une fonctionnelle quadratique.

Le réseau de neurones proposé est fondé sur le modèle discret de Hopfield. C'est un réseau de type récursif où les neurones sont totalement interconnectés. Cette méthodologie a l'avantage d'être souple lors de sa mise en oeuvre. Elle se résume aux deux points suivants :

- Codage du problème : il s'agit de ramener le problème à la minimisation de fonctionnelle (quadratique dans le cadre de ce mémoire)
- Mise en oeuvre de l'architecture neuronale pour la résolution de cette fonctionnelle.

Différents tests sur le comportement du réseau neuronal ont été effectués dans le cadre de l'estimation du mouvement, qui est un des problèmes les plus complets pour notre approche. Des résultats sur des séquences d'images de synthèse ont permis de montrer les avantages de notre méthodologie. On peut noter, en particulier, le rôle important du paramètre de quantification qui, dans le cas de séquences réelles, permet de mieux localiser la zone correspondant au mouvement de l'objet en limitant l'effet du lissage. Dans le cas d'images bruitées, le réseau semble être plus robuste dès lors que le bruit de l'image est plus important que celui de quantification généré par le réseau. La détermination de la valeur optimale de ce paramètre pour une configuration donnée peut être automatisée à partir de l'analyse des courbes d'énergie du réseau.

Par ailleurs, le réseau neuronal présente également un aspect intéressant du point de vue de son implémentation sous forme d'un circuit. En effet, la représentation de la variable solution dans un espace discret d'états de neurones offre une plus grande robustesse vis-à-vis d'un éventuel dysfonctionnement des neurones qui le constituent.

Une extension au contexte tridimensionnel ne pose pas de difficultés supplémentaires de par la méthodologie adoptée. Cette démarche consiste en effet à définir deux sous-réseaux supplémentaires pour représenter la troisième composante.

De plus, ce côté neuronal permet d'envisager, dans le contexte du mouvement, de prendre aisément en considération la notion de discontinuités sous forme d'un processus linéique, afin d'améliorer la qualité de la solution et d'obtenir une meilleure précision d'estimation. Cette prise en compte des discontinuités se fait au travers d'une simple modification des valeurs des poids synaptiques du réseau [Devlaminck, Lelaurin, Yacoubi, & Dubus, 1995].

Une évolution possible concerne également le modèle neuronal lui-même. Il semble envisageable de recourir au modèle de type Hopfield "étendu" dans lequel la fonction d'énergie est donnée par une expression d'ordre supérieur des sorties de neurones [Matsui & Nakabayashi, 1995]. Ce type de modèle constitue alors un outil efficace pour résoudre des problèmes d'optimisation plus complexes définis par des fonctionnelles d'ordre supérieur à deux.

Le travail relaté dans ce mémoire entrouvre une nouvelle voie pour la minimisation de fonctionnelles de type quadratique par le biais d'un réseau de neurones. Les différentes possibilités d'évolution de cette approche montrent bien les suites éventuelles à apporter à ce travail.

ANNEXES

ANNEXE I

L'apprentissage

Toute la connaissance dans un modèle connexionniste est codée dans son architecture et dans la valeur de ses poids synaptiques. L'apprentissage d'un réseau permet d'adapter ces valeurs, de rajouter ou de supprimer des connexions, pour répondre au problème posé. Plusieurs procédures d'apprentissage ont été et continuent à être développées. Ces différents algorithmes se répartissent en deux grandes catégories :

- l'apprentissage supervisé
- l'apprentissage non supervisé

A.I.1 Apprentissage supervisé

L'apprentissage supervisé nécessite la présence d'un "professeur" ou "superviseur" pour présenter au réseau un ensemble de couples entrée-sortie. Le réglage des poids est effectué par un algorithme itératif qui prend en compte l'erreur entre l'état de sortie désiré et l'état de sortie obtenu quand on présente une suite de modèles en entrée.

Les procédures relatives au réseau à couches sont le plus souvent supervisées. Ainsi, le perceptron de Rosenblatt ajustait les poids de chaque neurone 'i' par rapport à la réponse désirée d_i fournie par le professeur selon la règle suivante :

$$T_{ij}(t+1) = T_{ij}(t) + \alpha \cdot [d_i(t) - S_i(t)] \cdot V_j(t) \quad (\mathbf{I.1})$$

où $S_i(t)$ est l'état de sortie du neurone 'i' et α , le pas d'apprentissage. Cette procédure d'apprentissage converge si les données d'entrées sont linéairement séparables.

Widrow et Hoff ont pour leurs réseaux MADALINE (Multiple ADaptive LInear Elements) cherché à minimiser l'erreur quadratique (3.6) [Widrow, 1962] plutôt que l'erreur simple dans le cas du perceptron.

$$E = \frac{1}{2} \cdot \sum_{i=1}^N (d_i - S_i)^2 \quad (\mathbf{I.2})$$

Mais les limitations rencontrés par le perceptron, à savoir la séparabilité linéaire des classes, demeurent toujours vraies car seule la façon de déterminer les plans de séparation est améliorée. Il faudra attendre l'avènement de l'algorithme de rétropropagation du gradient [Rumelhart, Hinton, and Williams, 1986] [Lecun, 1987] et l'introduction de couches cachées pour surmonter cette limitation. Ainsi, dans un réseau à trois couches, la première couche permet de définir des hyperplans. La deuxième couche combine différents hyperplans et définit des hypercubes. La troisième couche réalise des combinaisons d'hypercube permettant ainsi de créer des surfaces frontières de forme complexe (voir Figure 1) [Lipmann, 1987].

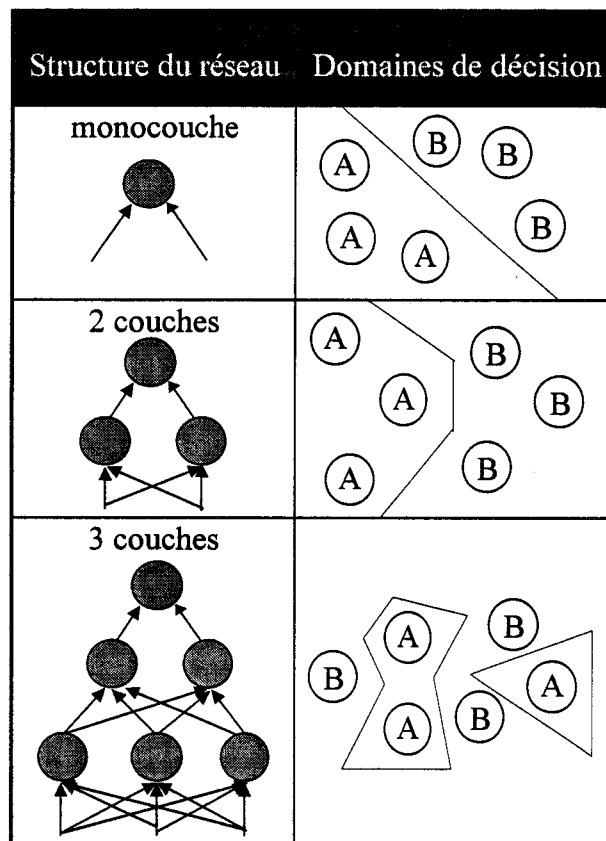


Figure 1 : Domaines de décision d'un perceptron multicouche

Par la suite, des améliorations de l'algorithme de rétropropagation du gradient ont été proposées [Hush & Horne, 1993] et Widrow, pour sa part, s'attacha à faire évoluer sa règle d'apprentissage pour les réseaux MADALINE [Widrow & Lehr, 1990].

A.I.2 Apprentissage non supervisé

Il n'existe pas de superviseur ou de professeur, le réseau s'auto-organise de façon à développer une certaine sélectivité aux caractéristiques des entrées. Ces algorithmes d'apprentissage sont sensés reproduire un mécanisme d'auto adaptation existant chez les êtres possédant un système nerveux [Rumelhart & Zipser, 1985] [Barlow, 1989].

La plupart de ces algorithmes sont basés sur la règle de Hebb [Hebb, 1949]. Bien qu'élaborée pour les réseaux biologiques, cette loi est l'un des premiers mécanismes d'évolutions des synapses :

"Quand une cellule A excite par son axone une cellule B et que, de manière répétée et persistante, elle participe à la genèse d'une impulsion dans B, un processus de croissance ou un changement métabolique a lieu dans l'une ou dans les deux cellules, de telle sorte que l'efficacité de A à déclencher une impulsion dans B est, parmi les autres cellules qui ont cet effet, accrue."

Selon Hebb, l'apprentissage est donc lié à une modification fonctionnelle dans les synapses. Ainsi, si deux neurones connectés entre eux sont actifs en même temps, la connexion qui les relie doit être renforcée (voir Figure 2.a). Dans le cas contraire (les deux neurones sont inactifs), elle n'est pas modifiée (voir Figure 2.b).

En revanche, si la cellule émettrice s'active sans que la cellule réceptrice le fasse, ou si la cellule réceptrice s'active en l'absence d'activité de la cellule émettrice, le poids de la synapse va s'affaiblir (voir Figure 2.c) car la connexion entre ces deux cellules n'est pas prépondérante dans le comportement de la cellule réceptrice.

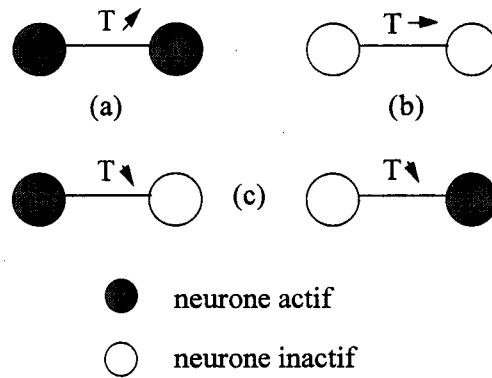


Figure 2 : La règle de Hebb

Ce comportement se traduit sous forme d'équation :

$$T_{ij}(t+1) = T_{ij}(t) + \mu \cdot V_i \cdot V_j \quad (\text{I.3})$$

où μ est une constante positive et représente le gain d'adaptation.

Malheureusement, le renforcement des synapses a lieu lorsque les neurones pré et post synaptiques sont simultanément activés ce qui provoque un risque d'instabilité. Diverses méthodes permettent de recourir à cette instabilité comme des inhibitions récurrentes au travers de synapses modifiables [Easton & Gordon, 1984], le seuillage, la normalisation des coefficients synaptiques [McClelland, 1986] ou encore l'apprentissage par compétition et principe du "winner take all" [Lipmann, 1987].

ANNEXE II

Phase d'Identification

1. Soit un problème d'optimisation décrit au moyen d'une forme quadratique de ses entrées, appelée fonctionnelle, dont le minimum correspond à la solution recherchée :

$$E = \sum_i \sum_j \left\{ \left[a_{i,j} \cdot z_{i,j} - y_{i,j} \right]^2 + \alpha^2 \cdot \left[\left(\sum_p \sum_q d_{p,q}^x \cdot z_{i+p,j+q} \right)^2 + \left(\sum_p \sum_q d_{p,q}^y \cdot z_{i+p,j+q} \right)^2 \right] \right\} \quad (\text{II.1})$$

où $z_{i,j}$ est la solution recherchée, $y_{i,j}$ sont les données du problème, $a_{i,j}$ des paramètres et, $d_{p,q}^x$ et $d_{p,q}^y$ représentent respectivement les coefficients des dérivées partielles en x et en y dans un domaine de définition \mathcal{D} comme l'illustre la Figure 1.

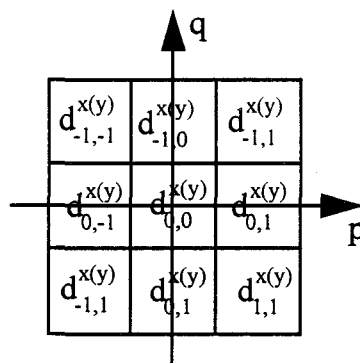


Figure 1 : représentation des coefficients $d_{p,q}^x$ ($d_{p,q}^y$) pour un domaine de définition \mathcal{D} de dimension 3×3

2. L'énergie du problème posé, l'approche développée consiste à représenter la variable solution $z_{i,j}$ en termes d'une somme d'états de neurones

$$z_{i,j} = k \cdot \sum_{h=1}^N V_{i,j,h} \quad (\text{II.2})$$

dans le but d'utiliser le processus de minimisation du réseau de type Hopfield discret caractérisé par la fonction d'énergie :

$$E = -\frac{1}{2} \sum_{i,j} \sum_{m,n} \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{i=1}^N \sum_{j=1}^N \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (\text{II.3})$$

3. Si l'on développe la relation (II.1), on obtient :

$$E = \sum_i \sum_j \left\{ a_{i,j}^2 \cdot z_{i,j}^2 + y_{i,j}^2 - 2 \cdot a_{i,j} \cdot y_{i,j} \cdot z_{i,j} + \alpha^2 \cdot \left[\sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x \cdot z_{i+p,j+q} \cdot z_{i+r,j+s} + \sum_p \sum_q \sum_r \sum_s d_{p,q}^y \cdot d_{r,s}^y \cdot z_{i+p,j+q} \cdot z_{i+r,j+s} \right] \right\} \quad (\text{II.4})$$

4. Nous allons à présent développer le second terme de la relation précédente en nous limitant au terme relatif à la dérivée partielle en x

$$\sum_i \sum_j \left[\sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x \cdot z_{i+p,j+q} \cdot z_{i+r,j+s} \right] \quad (\text{II.5})$$

5. On introduit $i'=i+p$ et $j'=j+q$, on a alors :

$$\sum_{i'} \sum_{j'} \left[\sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x \cdot z_{i'-p+r,j'-q+s} \cdot z_{i',j'} \right] \quad (\text{II.6})$$

6. Posons ensuite $S=\{(-p+r,-q+s)\}$, une série d'index et définissons la fonction :

$$f_{(i,j)\Theta_s} = \begin{cases} f_{i-p+r,j-q+s} & \text{si } p,q,r,s \in \mathcal{D} \\ 0 & \text{sinon.} \end{cases} \quad (\text{II.7})$$

7. En opérant de la même façon pour l'approximation de la dérivée partielle en y, on obtient comme nouvelle formulation pour (II.4) en substituant cette notation dans son second membre :

$$\begin{aligned}
E = & \sum_i \sum_j a_{i,j}^2 \cdot z_{i,j}^2 + \sum_i \sum_j y_{i,j}^2 + \sum_i \sum_j -2 \cdot a_{i,j} \cdot y_{i,j} \cdot z_{i,j} \\
& + \sum_i \sum_j \alpha^2 \cdot \left\{ \sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x \cdot z_{(i,j) \oplus s} \cdot z_{i,j} \right\} \\
& + \sum_i \sum_j \alpha^2 \cdot \left\{ \sum_p \sum_q \sum_r \sum_s d_{p,q}^y \cdot d_{r,s}^y \cdot z_{(i,j) \oplus s} \cdot z_{i,j} \right\}
\end{aligned} \tag{II.8}$$

8. Si l'on remplace à présent $z_{i,j}$ par sa nouvelle formulation (II.2), on aboutit, à la relation suivante :

$$\begin{aligned}
E = & \sum_{i,j} \sum_{m,n} \sum_h \sum_k a_{i,j}^2 \cdot \delta_{(i,j),(m,n)} \cdot V_{m,n,k} \cdot V_{i,j,h} + \sum_i \sum_j y_{i,j}^2 + \sum_i \sum_j \sum_h -2 \cdot a_{i,j} \cdot y_{i,j} \cdot V_{i,j,h} \\
& + \sum_{i,j} \sum_{m,n} \sum_h \sum_k \alpha^2 \cdot \left\{ \sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x \right\} \cdot \delta_{(i,j),(m,n) \oplus s} \cdot V_{(m,n) \oplus s,k} \cdot V_{i,j,h} \\
& + \sum_{i,j} \sum_{m,n} \sum_h \sum_k \alpha^2 \cdot \left\{ \sum_p \sum_q \sum_r \sum_s d_{p,q}^y \cdot d_{r,s}^y \right\} \cdot \delta_{(i,j),(m,n) \oplus s} \cdot V_{(m,n) \oplus s,k} \cdot V_{i,j,h}
\end{aligned} \tag{II.9}$$

9. En identifiant (II.9) avec (II.3) et en ignorant le terme constant

$$\sum_{i=1} \sum_{j=1} y_{i,j}^2 \tag{II.10}$$

on détermine les paramètres du réseau, poids des connexions et les entrées externes :

$$T_{i,j,h;m,n,k} = -2 \cdot k^2 \cdot \left\{ a_{i,j}^2 \cdot \delta_{(i,j),(m,n)} + \left(\sum_p \sum_q \sum_r \sum_s d_{p,q}^x \cdot d_{r,s}^x + d_{p,q}^y \cdot d_{r,s}^y \right) \cdot \delta_{(i,j),(m,n) \oplus s} \right\} \tag{II.11a}$$

$$I_{i,j,h} = -2 \cdot k \cdot a_{i,j} \cdot y_{i,j} \tag{II.11b}$$

où $\delta_{(i,j),(m,n)}$ représente la fonction de Kronecker définie par :

$$\delta_{(i,j),(m,n)} = \begin{cases} 1 & \text{si } (i,j) = (m,n) \\ 0 & \text{sinon} \end{cases} \tag{II.12}$$

ANNEXE III

Variation d'Energie

1. Rappelons la fonction d'énergie caractérisant le réseau :

$$E = -\frac{1}{2} \sum_{i=1}^L \sum_{j=1}^M \sum_{m=1}^L \sum_{n=1}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{i=1}^L \sum_{j=1}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (\text{III.1})$$

2. A partir de cette équation, la mise à jour des neurones s'effectue de manière asynchrone, autrement dit un à la fois. Soit le neurone (a,b,l) mis à jour. L'énergie (III.1) peut se reformuler en faisant apparaître les termes liés à l'état $V_{a,b,l}$ de ce neurone:

$$\begin{aligned} E &= -\frac{1}{2} \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{m=1}^L \sum_{\substack{n=1 \\ n \neq l}}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \\ &\quad - \frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{k=1}^N T_{a,b,l;m,n,k} \cdot V_{m,n,k} \cdot V_{a,b,l} - I_{a,b,l} \cdot V_{a,b,l} \\ &= -\frac{1}{2} \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{m=1}^L \sum_{\substack{n=1 \\ n \neq l}}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \\ &\quad - \frac{1}{2} \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{h=1}^N T_{i,j,h;a,b,l} \cdot V_{a,b,l} \cdot V_{i,j,h} - \frac{1}{2} \sum_{m=1}^L \sum_{n=1}^M \sum_{k=1}^N T_{a,b,l;m,n,k} \cdot V_{m,n,k} \cdot V_{a,b,l} - I_{a,b,l} \cdot V_{a,b,l} \end{aligned} \quad (\text{III.2})$$

3. Puisque la matrice des connexions est définie symétrique, $T_{i,j,h;m,n,k} = T_{m,n,k;i,j,h}$, on peut réécrire cette relation sous la forme suivante :

$$\begin{aligned} E &= -\frac{1}{2} \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{m=1}^L \sum_{\substack{n=1 \\ n \neq l}}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{\substack{i=1 \\ i \neq a}}^L \sum_{\substack{j=1 \\ j \neq b}}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \\ &\quad - \sum_{m=1}^L \sum_{n=1}^M \sum_{k=1}^N T_{a,b,l;m,n,k} \cdot V_{m,n,k} \cdot V_{a,b,l} - I_{a,b,l} \cdot V_{a,b,l} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l} \cdot V_{a,b,l} \end{aligned} \quad (\text{III.3})$$

4. Or, on sait que :

$$u_{a,b,l} = \sum_m^L \sum_n^M \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} + I_{a,b,l} \quad (\text{III.4})$$

5. Si l'on reporte dans (III.3), on obtient :

$$E = -\frac{1}{2} \sum_{i=1}^L \sum_{j=1}^M \sum_{m=1}^L \sum_{n=1}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{i=1}^L \sum_{j=1}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (\text{III.5})$$

$$- u_{a,b,l} \cdot V_{a,b,l} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l} \cdot V_{a,b,l}$$

6. Soit $V_{a,b,l}^{\text{old}}$ l'état actuel du neurone l du sous-réseau localisé en (a,b) et $V_{a,b,l}^{\text{new}}$, son nouvel état défini par :

$$V_{a,b,l}^{\text{new}} = H(u_{a,b,l}^{\text{old}}) \quad (\text{III.6})$$

où H est la fonction de Heaviside.

7. Les énergies E^{old} et E^{new} correspondant respectivement à chacun des deux états $V_{a,b,l}^{\text{old}}$ et $V_{a,b,l}^{\text{new}}$ s'écrivent suite à la relation (III.5) :

$$E^{\text{old}} = -\frac{1}{2} \sum_{i=1}^L \sum_{j=1}^M \sum_{m=1}^L \sum_{n=1}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{i=1}^L \sum_{j=1}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (\text{III.7a})$$

$$- u_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}}$$

$$E^{\text{new}} = -\frac{1}{2} \sum_{i=1}^L \sum_{j=1}^M \sum_{m=1}^L \sum_{n=1}^M \sum_{h=1}^N \sum_{k=1}^N T_{i,j,h;m,n,k} \cdot V_{m,n,k} \cdot V_{i,j,h} - \sum_{i=1}^L \sum_{j=1}^M \sum_{h=1}^N I_{i,j,h} \cdot V_{i,j,h} \quad (\text{III.7b})$$

$$- u_{a,b,l}^{\text{new}} \cdot V_{a,b,l}^{\text{new}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{new}} \cdot V_{a,b,l}^{\text{new}}$$

8. Définissons la variation d'énergie ΔE subit par le réseau suite à ce changement d'état :

$$\Delta E = E^{\text{new}} - E^{\text{old}} \quad (\text{III.8a})$$

$$\begin{aligned} \Delta E = & -u_{a,b,l}^{\text{new}} \cdot V_{a,b,l}^{\text{new}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{new}} \cdot V_{a,b,l}^{\text{new}} \\ & - \left(-u_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}} \right) \end{aligned} \quad (\text{III.8b})$$

9. Suite au nouvel état $V_{a,b,l}^{\text{new}}$, le nouveau potentiel $u_{a,b,l}^{\text{new}}$ peut s'écrire en fonction du potentiel actuel du neurone $u_{a,b,l}^{\text{old}}$:

$$u_{a,b,l}^{\text{new}} = u_{a,b,l}^{\text{old}} + T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{new}} \quad (\text{III.9})$$

10. En reportant dans la relation (III.8b), la variation d'énergie s'exprime alors en fonction des données disponibles :

$$\begin{aligned} \Delta E = & - \left(u_{a,b,l}^{\text{old}} + T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{new}} \right) \cdot V_{a,b,l}^{\text{new}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{new}} \cdot V_{a,b,l}^{\text{new}} \\ & - \left(-u_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}} + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot V_{a,b,l}^{\text{old}} \cdot V_{a,b,l}^{\text{old}} \right) \\ = & - \left(u_{a,b,l}^{\text{old}} \cdot (V_{a,b,l}^{\text{new}} - V_{a,b,l}^{\text{old}}) + \frac{1}{2} \cdot T_{a,b,l;a,b,l} \cdot (V_{a,b,l}^{\text{new}^2} + V_{a,b,l}^{\text{old}^2}) \right) \end{aligned} \quad (\text{III.10})$$

11. Posons $\Delta V_{a,b,l}$ la variation du vecteur d'état due à la transition du neurone (a,b,l) par :

$$\Delta V_{a,b,l} = V_{a,b,l}^{\text{new}} - V_{a,b,l}^{\text{old}} \quad (\text{III.11})$$

12. En raison de la valeur de $V_{a,b,l}^{\text{old}}$ et du potentiel $u_{a,b,l}^{\text{old}}$, quatre configurations sont possibles :

$V_{a,b,l}^{\text{old}}$	0	1
$u_{a,b,l}^{\text{old}} > 0 \Rightarrow V_{a,b,l}^{\text{new}} = 1$	$\Delta V_{a,b,l} = 1$	$\Delta V_{a,b,l} = 0$
$u_{a,b,l}^{\text{old}} < 0 \Rightarrow V_{a,b,l}^{\text{new}} = 0$	$\Delta V_{a,b,l} = 0$	$\Delta V_{a,b,l} = -1$

Tableau 1 : Valeurs de $\Delta V_{a,b,l}$

13. Le Tableau 1 montre, lorsqu'il y a changement d'état ($\Delta V_{a,b,l} \neq 0$), que le produit $u_{a,b,l}^{\text{old}} \cdot \Delta V_{a,b,l}$ est toujours positif ce qui conduit, dans le cas où $T_{a,b,l;a,b,l}$ est positif ou nul, pour la relation (III.10) à :

$$\Delta E \leq 0 \quad (\text{III.12})$$

14. Dans le cas contraire où $T_{a,b,l;a,b,l} < 0$ et $|T_{a,b,l;a,b,l}| > 2 \cdot |u_{a,b,l}^{\text{old}}|$, alors

$$\Delta E \geq 0 \quad (\text{III.13})$$

De ce fait, pour s'assurer la convergence du réseau, une règle de décision déterministe ou stochastique est nécessaire.

15. L'expression générale de la variation d'énergie ΔE due à une transition d'un neurone est :

$$\Delta E = - \left(u_{i,j,h} \cdot \Delta V_{i,j,h} + \frac{1}{2} \cdot T_{i,j,h;i,j,h} \cdot (\Delta V_{i,j,h})^2 \right) \quad (\text{III.14})$$

ANNEXE IV

Approche de Horn & Schunck

1. Définition de la fonctionnelle du flot optique

$$E = \iint \left\{ (f_x \cdot w_x + f_y \cdot w_y + f_t)^2 + \alpha^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_x}{\partial y} \right)^2 + \left(\frac{\partial w_y}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 \right] \right\} \cdot dx \cdot dy \quad (\text{IV.1})$$

2. Posons,

$$F = \left\{ (f_x \cdot w_x + f_y \cdot w_y + f_t)^2 + \alpha^2 \cdot \left[\left(\frac{\partial w_x}{\partial x} \right)^2 + \left(\frac{\partial w_x}{\partial y} \right)^2 + \left(\frac{\partial w_y}{\partial x} \right)^2 + \left(\frac{\partial w_y}{\partial y} \right)^2 \right] \right\} \quad (\text{IV.2})$$

3. D'après les équations d'Euler-Lagrange, on peut écrire pour chaque indice i :

$$\frac{\partial F}{\partial w_i} - \frac{\partial}{\partial x} \cdot \left(\frac{\partial F}{\partial p} \right) - \frac{\partial}{\partial y} \cdot \left(\frac{\partial F}{\partial q} \right) = 0$$

(IV.3)

$$\text{avec} \quad \begin{cases} \frac{\partial w_i}{\partial x} = p \\ \frac{\partial w_i}{\partial y} = q \end{cases}$$

4. En résolvant ces deux équations, on aboutit :

$$2 \cdot f_x \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t) - \alpha^2 \cdot \left(2 \cdot \frac{\partial^2 w_x}{\partial^2 x} + 2 \cdot \frac{\partial^2 w_x}{\partial^2 y} \right) = 0$$

(IV.4)

$$2 \cdot f_y \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t) - \alpha^2 \cdot \left(2 \cdot \frac{\partial^2 w_y}{\partial^2 x} + 2 \cdot \frac{\partial^2 w_y}{\partial^2 y} \right) = 0$$

5. que l'on peut reformuler sous l'écriture suivante :

$$\alpha^2 \cdot \nabla^2 w_x = f_x \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t) \quad (\text{IV.5})$$

$$\alpha^2 \cdot \nabla^2 w_y = f_y \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t)$$

6. Si l'on approxime les laplaciens par :

$$\nabla^2 w_x \approx \kappa \cdot (\bar{w}_x - w_x) \quad (\text{IV.6})$$

$$\nabla^2 w_y \approx \kappa \cdot (\bar{w}_y - w_y)$$

avec

$$\begin{aligned} \bar{w}_{(i,j)x} &= \frac{1}{6} \cdot (w_{(i-1,j)x} + w_{(i,j-1)x} + w_{(i,j+1)x} + w_{(i+1,j)x}) \\ &\quad + \frac{1}{12} \cdot (w_{(i-1,j-1)x} + w_{(i-1,j+1)x} + w_{(i+1,j-1)x} + w_{(i+1,j+1)x}) \end{aligned} \quad (\text{IV.7})$$

$$\begin{aligned} \bar{w}_{(i,j)y} &= \frac{1}{6} \cdot (w_{(i-1,j)y} + w_{(i,j-1)y} + w_{(i,j+1)y} + w_{(i+1,j)y}) \\ &\quad + \frac{1}{12} \cdot (w_{(i-1,j-1)y} + w_{(i-1,j+1)y} + w_{(i+1,j-1)y} + w_{(i+1,j+1)y}) \end{aligned}$$

7. et que l'on reporte dans (IV.5), on a¹ :

$$\alpha^2 \cdot (\bar{w}_x - w_x) = f_x \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t) \quad (\text{IV.8})$$

$$\alpha^2 \cdot (\bar{w}_y - w_y) = f_y \cdot (f_x \cdot w_x + f_y \cdot w_y + f_t)$$

8. Si l'on sépare les termes relatifs à w_x et w_y des autres termes, on obtient :

$$\alpha^2 \cdot \bar{w}_x - f_x \cdot f_t = f_x \cdot (f_x \cdot w_x + f_y \cdot w_y) + \alpha^2 \cdot w_x \quad (\text{IV.9})$$

$$\alpha^2 \cdot \bar{w}_y - f_y \cdot f_t = f_y \cdot (f_x \cdot w_x + f_y \cdot w_y) + \alpha^2 \cdot w_y$$

¹ Remarque : le coefficient κ est intégré dans le paramètre α .

9. En réécrivant (IV.9) sous forme matricielle,

$$\begin{bmatrix} f_x^2 + \alpha^2 & f_x \cdot f_y \\ f_y \cdot f_x & f_y^2 + \alpha^2 \end{bmatrix} \cdot \begin{bmatrix} w_x \\ w_y \end{bmatrix} = \begin{bmatrix} \alpha^2 \cdot \bar{w}_x - f_x \cdot f_t \\ \alpha^2 \cdot \bar{w}_y - f_y \cdot f_t \end{bmatrix} \quad (\text{IV.10})$$

10. La résolution de cet équation matricielle donne :

$$\begin{bmatrix} w_x \\ w_y \end{bmatrix} = \frac{1}{\Delta} \cdot \begin{bmatrix} f_y^2 + \alpha^2 & -f_y \cdot f_x \\ -f_x \cdot f_y & f_x^2 + \alpha^2 \end{bmatrix} \cdot \begin{bmatrix} \alpha^2 \cdot \bar{w}_x - f_x \cdot f_t \\ \alpha^2 \cdot \bar{w}_y - f_y \cdot f_t \end{bmatrix} \quad (\text{IV.11a})$$

où

$$\Delta = (\alpha^2 + f_x^2) \cdot (\alpha^2 + f_y^2) - f_x^2 \cdot f_y^2 = \alpha^2 \cdot (\alpha^2 + f_x^2 + f_y^2) \quad (\text{IV.11b})$$

11. Si l'on développe le produit du membre de droite de la relation (IV.10), on aboutit à la relation matricielle :

$$\begin{bmatrix} w_x \\ w_y \end{bmatrix} = \frac{\alpha^2}{\Delta} \cdot \begin{bmatrix} (f_y^2 + \alpha^2) \cdot \bar{w}_x - f_y \cdot f_x \cdot \bar{w}_y - f_x \cdot f_t \\ (f_x^2 + \alpha^2) \cdot \bar{w}_y - f_y \cdot f_x \cdot \bar{w}_x - f_y \cdot f_t \end{bmatrix} \quad (\text{IV.12})$$

12. Si l'on réécrit cette relation sous la forme suivante :

$$\begin{bmatrix} (\alpha^2 + f_x^2 + f_y^2) & 0 \\ 0 & (\alpha^2 + f_x^2 + f_y^2) \end{bmatrix} \cdot \begin{bmatrix} w_x - \bar{w}_x \\ w_y - \bar{w}_y \end{bmatrix} = \begin{bmatrix} -f_x \cdot (f_x \cdot \bar{w}_x + f_y \cdot \bar{w}_y + f_t) \\ -f_y \cdot (f_x \cdot \bar{w}_x + f_y \cdot \bar{w}_y + f_t) \end{bmatrix} \quad (\text{IV.13})$$

13. La solution itérative devient alors :

$$w_x^{n+1} = \bar{w}_x^n - f_x \cdot (f_x \cdot \bar{w}_x^n + f_y \cdot \bar{w}_y^n + f_t) / (\alpha^2 + f_x^2 + f_y^2)$$

(IV.14)

$$w_y^{n+1} = \bar{w}_y^n - f_y \cdot (f_x \cdot \bar{w}_x^n + f_y \cdot \bar{w}_y^n + f_t) / (\alpha^2 + f_x^2 + f_y^2)$$

Bibliographie

- [Abu-Mostafa & Saint Jacques, 1985] "Information capacity of the Hopfield network," *I.E.E.E. Transaction on Information Theory*, 31, pp 461-464.
- [Ackley, Hinton & Sejnowsky, 1985] "A learning algorithm for Boltzmann machine," *Cognitive Science*, 9, pp 147-169.
- [Aggarwal, Davis, & Martin, 1981] "Correspondance processes in dynamic scene analysis," *Proceedings of the I.E.E.E.*, 69(5), pp 562-572.
- [Aggarwal & Duda, 1975] "Computer analysis of moving polygonal images," *I.E.E.E. Transactions on Computer*, 24(10), pp 966-976.
- [Amari, 1972] "Learning patterns and pattern sequences by self organizing nets of threshold elements," *I.E.E.E. Transaction on Computers*, 21, pp 1197-1206.
- [Amit & Gutfreund, 1985] "Spin-glass models of neural networks," *Physical Review A*, 32, pp 1007-1018.
- [Amit, 1987] "The properties of models of simple neural networks," Springer Verlag, Berlin, pp 430-484.
- [Anderson, 1972] "A simple neural network generating interactive memory," *Mathematical Bioscience*, 14, pp 197-220.

- [Anderson, Silverstein, Ritz, and Jones, 1977] "Distinctive features, categorical perception, and probability learning : some applications of a neural model," *Psychological Review*, 84, pp 413-451, and reprinted in *Neurocomputing*, Anderson, J.A. and Rosenfeld, E. (1988). MIT Press, Cambridge, MA.
- [Anderson & Rosenfeld, 1988] *Neurocomputing : Foundations of Research*, MIT Press, Cambridge, MA.
- [Andrews & Hunt, 1977] *Digital Image Restoration*, Prentice-Hall, Englewood Cliffs, NJ.
- [Aubin, Nepomiastchy, & Charles, 1982] *Méthodes explicites de l'optimisation*, Dunod, Paris.
- [Badeva & Morozov, 1991] *Problèmes incorrectement posés*, Masson, Paris.
- [Barlow, 1989] "Unsupervised Learning," *Neural Computation*, 1, pp 295-311.
- [Barnard & Thompson, 1980] "Disparity analysis of images," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 2(4), pp 333-340.
- [Barnett & Storey, 1970] *Matrix methods in stability theory*, Th. Nelson & Sons, London.
- [Barron & Barron, 1988] "Statistical learning networks : A unifying view," *Symposium on the Interface: Statistics and Computing Science*, Reston, VA, pp 192-203.
- [Barrow & Tennenbaum, 1981] "Interpreting Line Drawings as three-dimensional Surfaces," *Artificial Intelligence*, 17, pp 75-117.
- [Bilbro & Snyder, 1988] "Range image restauration using mean field annealing," *I.E.E.E. Transaction on Neural Information Processing systems*, Denver, CO.
- [Block, 1962] "The Perceptron : a model of brain functioning. I," *Reviews of Modern Physics*, 34, pp 123-135.
- [Bout & Miller, 1989] "Graph partitionning using annealed neural networks," *International Joint Conference on Neural network*, Washington, DC.
- [Bouthemy, 1989] "A maximum-likelihood framework for determining moving edges," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 11(5), pp 499-511.

-
- [Brady & Yuille, 1984] "An extremum principle for shape from contour," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 6, pp 288-301.
- [Burt, Yen & Xu, 1982] "Local correlation measures for motion analysis : a comparative study," *Proceedings of the Conference on Pattern Recognition and Image Processing*, Las Vegas, pp 269-274.
- [Carpenter & Grossberg, 1987] "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Computer Vision, Graphics and Image Processing*, 37, pp 54-115.
- [Cea, 1971] *Optimisation, Théorie et Algorithmes*, Dunod, Paris.
- [Ciarlet, 1988] *Introduction à l'analyse numérique matricielle et à l'optimisation*, Masson, Paris.
- [Cheng & Huang, 1984] "Image registration by matching relational structures," *Pattern Recognition*, 17(1), pp 149-159.
- [Cohen & Grossberg, 1983] "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks," *I.E.E.E. Transaction on Systems, man and Cybernetics*, 13, pp 815-826.
- [Cric & Asanuma, 1986] "Certain aspects of the anatomy and physiology of the cerebral cortex," in *Parallel Distributed Processing*, vol. 2, chap. 20, MIT Press, Cambridge, MA.
- [DARPA, 1988] *DARPA Neural Network Study*, A.F.C.E.A. International Press, Fairfax, VA.
- [Daugmann, 1988] "Networks for image analysis : Motion and Texture," *Proceedings of the First international Joint Conference on Neural Networks*, Washington D.C., pp 189-193.
- [Davallo & Naim, 1990] *Des Réseaux de Neurones*, Paris-Eyrolles.
- [Dembo, 1989] "On the capacity of associative memories with linear threshold functions," *I.E.E.E. Transaction on Information Theory*, 35, pp 709-720.

-
- [Devlaminck & Dubus, 1996] "Estimation of compressible or incompressible deformable motions for density images," *Soumis à I.E.E.E. International Conference on Image Processing*, Lausanne, Switzerland.
- [Durbin & Rumelhart, 1989] "Product units : A Computationally Powerfull and Biologically Plausible Extension to Backpropagation Networks," *Neural Computation*, 1, pp 133-142.
- [Easton & Gordon, 1984] "Stabilization of Hebbian Neural Nets by Inhibitory Learning," *Biological Cybernetics*, 29, pp 127-136.
- [Edelman, 1987] *Neural Darwinism : The Theory of Neuronal Group Selection*, Basic, New-York.
- [Enkelmann, 1986] "Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences," *Workshop on motion : Rep. And Analysis*, Charleston, pp 81-87.
- [Farley & Clark, 1954] "Simulation of self-organizing systems by digital computer," *IRE Transaction of Information Theory*, 4, pp 76-84.
- [Fennema & Thompson, 1979] "Velocity determination in scenes containing several moving objects," *Computer Vision, Graphics and Image Processing*, 9(4), pp 301-315.
- [Fleet & Jepson, 1990] "Computation of normal velocity from local phase information," *Proceedings of I.E.E.E. Conference on Computer Vision and Pattern Recognition*, San Diego, pp 379-386.
- [Fitzpatrick, 1985] "A method for calculating fluid optical flow in time depend image based on the continuity equation," *Proceedings of the I.E.E.E. Conference on Computer Vision and Pattern Recognition*, 85, San Francisco, CA, pp 78-81.
- [Fitzpatrick, 1988] "The existence of geometrical density-image transformation corresponding to object motion," *Computer Vision, Graphics and Image Processing*, 44, pp 155-174.
- [Fogelman-Soulie & Weisbuch, 1987] "Random iterations of threshold networks and associative memory," *S.I.A.M. Journal Computing*, 16, pp 203-220.

- [Fogelman-Soulie, 1985] "Contribution à une théorie du calcul sur réseau," *Thèse de doctorat*, Université de Grenoble.
- [Fukushima, 1975] "Cognitron : A self-organizing multilayered Neural Network," *Biological Cybernetics*, 20, pp 121-136.
- [Geman & Geman, 1984] "Stochastic relaxation, Gibbs distributions and the bayesian restoration of images, *I.E.E.E. Proceedings of Pattern Analysis and Machine Intelligence*, 6, pp721-741.
- [Gidas, 1985] "Non-stationary Markov chains and convergence of the annealing algorithms," *Journal of Statistical Physics*, 39, pp 73.
- [Golub & Van Loan, 1983] *Matrix Computations*, North Oxford Academic, Oxford.
- [Grimson, 1981] *From images to surfaces : a computational study of the human early visual system*, MIT Press, Cambridge, MA.
- [Grossberg, 1976] "Adaptive pattern classification and universal recoding : II. feedback, expectation, olfaction, illusions," *Biological Cybernetics*, 23, pp 187-202.
- [Grossberg & Mingolla, 1985] "Neural dynamics of form perception : Boundary completion, illusory figures, and neon color spreading," *Psychological Review*, 92, pp 173-211.
- [Hadamard, 1902] "Sur les problèmes aux dérivées partielles et leur signification physique," *Bulletin of University of Princeton*, 13 (49).
- [Hadamard, 1932] *Le problème de Cauchy aux dérivées partielles et leur signification physique*, Hermann, Paris.
- [Hamman & Ocio, 1987] "Les verres de spin et l'étude des milieux désordonnés," *Pour la Science*, pp 40-50.
- [Hartman, Keeler, and Kowalski, 1990] "Layered neural networks with Gaussian hidden units as universal approximations," *Neural Computation*, 2(2), pp 210-215.
- [Hebb, 1949] *The Organization of Behavior*, Wiley, New-York.
- [Hecht-Nielsen, 1987] "Counter-propagation networks," *Applied Optics*, 26, pp 4979-4984.

- [Hecht-Nielsen, 1990] *Neurocomputing*, HNC Inc. & University of California, Addison Wesley Publishing Company, San Diego, CA.
- [Heeger, 1988] "Optical flow using spatio-temporal filters," *International Journal of Computer Vision*, 1(4), pp 279-302.
- [Heitz & Bouthemy, 1990] "Multimodal motion estimation and segmentation using markov random fields," *Proceedings of the 10th I.E.E.E. Conference on Pattern Recognition*, Atlantic City, pp 378-383.
- [Heitz & Bouthemy, 1993] "Multimodal estimation of discontinuous optical flow using markov random fields," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 15(12), pp 1217-1232.
- [Herault, 1989] "Réseaux de neurones récurrents pour l'optimisation combinatoire : Application à la théorie des graphes et à la vision par ordinateur," *Thèse de Doctorat*, Université de Grenoble.
- [Hest, 1975] *Optimization theory*, Wiley & Sons, New-York.
- [Hildreth, 1984a] "Computation of the velocity field," *Proceedings of Royal Society of London B*, 221, pp 189-220.
- [Hildreth, 1984b] *The measurement of visual motion*, MIT Press, Cambridge, MA.
- [Hinton, Sejnowski, and Ackley, 1984] "Boltzmann Machines : Constraint satisfaction networks that learn," *Carnegie Mellon University Technical Report*, CMU-CS-84-119.
- [Hopfield, 1982] "Neural networks and physical systems with emergent collective computational abilities," *Proceedings of National Academic Sciences U.S.A.*, 79, pp 2554-2558.
- [Hopfield, 1984] "Neurons with graded response have collective computational properties like those of two-state neurons," *Proceedings of National Academic Sciences U.S.A.*, 81, pp 3088-3092.
- [Hopfield & Tank, 1985] "Neural Computation of decisions in optimization problems," *Biological cybernetics*, 52, pp 141-152.

- [Hopfield & Tank, 1986] "Simple Neural Optimization Networks : An A/D Converter, Signal Decision Circuit, and a Linear Programming Circuit," *I.E.E.E. Transactions on Circuits and Systems*, 33, pp 533-541.
- [Horn, 1974] "Determining lightness from an image," *Computer Vision, Graphics and Image Processing*, 3, pp 111-299.
- [Horn & Schunk, 1981] "Determining optical flow," *Artificial Intelligence*, 17, pp 185-203.
- [Hush & Horne, 1993] "Progress in Supervised Neural Networks," *I.E.E.E. Signal Processing Magazine*, pp 8-39.
- [Ikeuchi, 1980a] "Numerical shape from shading and occluding contours in a single view," *memo n°556*, Artificial Intelligence Laboratory, MIT, Cambridge, MA.
- [Ikeuchi, 1980b] "Shape from regular patterns," *memo n°556*, Artificial Intelligence Laboratory, MIT, Cambridge, MA.
- [Ikeuchi & Horn, 1981] "Numerical shape from shading and occluding boundaries," *Artificial Intelligence*, 17, pp 141-184.
- [Ising, 1925] "Beitrag zur Theorie des Ferromagnetismus," *Z. Physik*, 31, pp 253-258.
- [Ivanov, 1962] "Sur les tâches linéaires incorrectement posées," *Dokl. AN SSSR*, 145(2), pp 270.
- [Jacobson & Wechsler, 1988] "Derivation of optical flow using a spatio-temporal frequency approach," *Computer Vision, Graphics and Image Processing*, 38, pp 29-65.
- [Kac, 1966] "Can one hear the shape of a drum?," *Am. Math. Monthly*, 73(4), pp 1-23.
- [Kamp & Hasler, 1990] *Réseaux de neurones récurrents pour mémoires associatives*, Presses polytechniques et universitaires romandes.
- [Kirkpatrick, Gelatt & Vecchi, 1983] "Optimization by simulated annealing," *Science*, 220, pp 671-680.
- [Kirkpatrick, 1984] "Optimization by simulated annealing : quantitative studies," *Journal of Statistical Physics*, 34, pp 974.
- [Kirkpatrick & Sherrington, 1978] "Infinite ranged models of spins-glasses," *Physical Review B*, 17, pp 4384.

-
- [Klopf, 1972] "Brain function and adaptive systems : A heterostatic theory," *Air Force Research Laboratories Technical Report*, AFCRL-72-0164.
- [Koch & Ullman, 1985] "Shifts in selective visual attention : towards the underlying neural circuitry", *Human Neurobiology*, 4, pp 219-227.
- [Kohonen, 1972] "Correlation matrix memories," *I.E.E.E. Transaction on Computers*, 21, pp 353-359.
- [Kohonen, 1984] *Self-Organization and associative memory*, Springer-Verlag, 3^e édition.
- [Konrad & Dubois, 1990] "A comparison of stochastic and deterministic solution methods in bayesian estimation of 2D motion," *Proceedings of the first European Conference on Computer Vision*, Antibes, pp 149-160.
- [Kosko, 1987] "Adaptive bidirectional associative memories," *Applied Optics*, 24, pp 4947-4960.
- [Laarhoven, 1987] *Simulated annealing : theory and applications*, D. Reidel Publishing Company.
- [LaSalle, 1986] *The Stability and Control of Discrete Processes*, Springer-Verlag, New-York.
- [Lavrentiev, 1955] "Sur le problème de Cauchy concernant l'équation de Laplace," *Dokl. AN SSSR*, 102(2), pp 205.
- [Lecun, 1987] "Modèles connexionnistes de l'apprentissage," *Thèse de Doctorat*, Université de Paris IV.
- [Lee & Kil, 1991] "A gaussian potential function network with hierarchically self-organizing learning," *Neural Network*, 4, pp 207-224.
- [Lemberg, 1988] "Les réseaux neuronaux artificiels," *Langages & Systèmes, info P.C.*, 71.
- [Lin, Liao, Tsao, & Lingutla, 1991] "A hierarchical multiple view approach to 3D object recognition," *I.E.E.E. Transaction on Neural Network*, 2(1), pp 84-90.
- [Lipmann, 1987] "An introduction to computing with neural nets," *I.E.E.E. A.S.S.P. Magazine*, pp 4-23.
- [Marr, 1982] *Vision*, Freeman, San Francisco.

- [Marr & Ullman, 1981] "Directional selectivity and its use in early visual processing," *Proceedings of Royal Society of London B*, 211(1183), pp 151-180.
- [Marren, Harston, & Pap, 1991] *Handbook of Neural Computing Applications*, Academic Press Inc., Sna Diego, CA.
- [Matsui & Nakabayashi, 1995] "Minimum searching by extended Hopfield Model," *I.E.E.E. International Conference on Neural Networks (I.C.N.N.'95)*, Perth, Western Australia, 5, pp 2648-2651.
- [McClelland & Rumelhart, Hinton, 1986] "The Appeal of Parallel Distributed Processing," *Parallel Distributed Processing*, vol. I, chap. 1, Editors Rumelhart, D.E., and McClelland, J.L. and the PDP Research Group (Eds.). MIT Press, Cambridge, MA.
- [McCulloch & Pitts, 1943] "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biophysics*, 5, pp 115-133.
- [McEliece, Postner, Rodenmich, & Venkatesh, 1987] "The capacity of the Hopfield associative memory," *I.E.E.E. Transaction on Information Theory*, 33, pp 461-482.
- [Mead, 1989] *Analog VLSI and Neural Systems*, Addison-Wesley Publishing Co.
- [Metropolis, Rosenbluth, Rosenbluth, Teller, & Teller, 1953] "Equation of state calculations for fast computing machines," *Journal of chemical Physics*, 6, pp 1087.
- [Minsky & Papert, 1969] *Perceptrons*, MIT Press, Cambridge, MA.
- [Nagel, 1987] "On the estimation of optical flow : relations between different approaches and some new results," *Artificial Intelligence*, 33, pp 299-324.
- [Nastar, 1994] "Modèles physiques déformables et modes vibratoires pour l'analyse du mouvement non-rigide dans les images multidimensionnelles," *Thèse de Doctorat*, Ecole Nationale des Ponts et Chaussées.
- [Nilsson, 1965] *Learning Machines : Foundations of Trainable Pattern-Classifying Systems*, McGraw-Hill, New-York.

- [Page & Tagliarini, 1988] "Algorithm development for neural networks," *S.P.I.E. Symposium on Innovative Science and Technology*, Los Angeles, CA.
- [Parker, 1982] "Learning-logic," *Invention Report S81-64, File 1*, Office of Technology Licensing, Stanford University, CA.
- [Parker, 1985] "Learning-logic," *Technical Report TR-47*, Center for Computational Research in Economics and Management Science, MIT, Cambridge, MA.
- [Peretto, 1987] "Collective properties of neural networks : a statistical physics approach," *Biological cybernetics*, 50, pp 51-64.
- [Personnaz, Guyon, & Dreyfus, 1986] "Collective computational properties of neural networks : new learning mechanism," *Physical Review A*, 34, pp 4217-4228.
- [Phillips, 1962] "Technique for the numerical solution of integral equations of the first kind," *J. Assoc. Comput. Mach.*, 9(1), pp 84.
- [Pineda, 1987] "Generalization of back propagation to recurrent Neural Networks," *Physical Review Letter*, 19, pp 2229-2232.
- [Poggio & Koch, 1985] "Ill-posed problems in early vision : from computational theory to analogue networks," *Proceedings of Royal Society of London B*, 226(1244), pp 303-323.
- [Poggio, Torre, & Koch, 1985] "Computational vision and regularization theory," *Nature*, 317, pp 314-319.
- [Poggio, Voorhees, & Yuille, 1985] "Regularizing edge detection," *memo n°776*, Artificial Intelligence Laboratory, MIT, Cambridge, MA.
- [Powell, 1985] "*Radial Basis Functions for multivariate interpolation : A review*," Technical Report DAMPT 1985/NA12, Department of Applied Mathematics and Theoretical Physics, Cambridge University, Cambridge, England.
- [Proceedings of I.E.E.E., 1990] "Neural Networks I and II," 78, (9) (10).
- [Pun, Gerig, & Ratib, 1994] "Image Analysis and Computer Vision in medicine," *Computerized Medical Imaging and Graphics*, 18(2), pp 85-96.
- [Rémy, 1994] *l'Intelligence Artificielle*, Dunod, Paris.

- [Rosenblatt, 1958] "The Perceptron : A probabilistic model for information storage and organization in the brain," *Psychological Review*, 65, pp 386-408, and reprinted in *Neurocomputing*, Anderson, J.A. and Rosenfeld, E. (1988). MIT Press, Cambridge, MA, pp 92-114.
- [Rosenblatt, 1961] *Principles of Neurodynamics : perceptrons and the theory of brain mechanisms*, Spartan Books, Washington.
- [Rougée, Levy & Willsky, 1987] "Reconstruction of two-dimensional velocity fields as a linear estimation problem," *Proceedings of the first International Conference on Computer Vision*, London, pp 646-650.
- [Rumelhart & Zipser, 1985] "Feature Discovery by Competitive Learning," *Cognitive Science*, 9, pp 75-112.
- [Rumelhart & McClelland, 1986] *Parallel Distributed Processing*, vol. I and II, MIT Press, Cambridge, MA.
- [Rumelhart, Hinton, and Williams, 1986] "Learning internal representations by error propagation," *Parallel Distributed Processing*, vol. I, chap. 8, pp 318-364, Editors Rumelhart, D.E., and McClelland, J.L. and the PDP Research Group (Eds.). MIT Press, Cambridge, MA.
- [Samy & Bozzo, 1984] "Extension de la transformée de Hough au traitement d'une séquence d'images," *Proceedings du premier colloque Image CESTA-GRETSI*, Biarritz, 1984.
- [Schunck, 1986] "The Image flow constraint equation," *Computer vision, Graphics and Image Processing*, 35, pp 20-46.
- [Sejnowski ,1986] "Open questions about computation in cerebral cortex," in *Parallel Distributed Processing*, vol. 2, chap. 21, MIT Press, Cambridge, MA.
- [Shirai, 1987] "Three-Dimensional Computer Vision," *Springer Series in Symbolic Computation*, chap. 7, Springer-Verlag.
- [Song & Leahy, 1991] "Computation of 3D velocity fields from 3D cine CT camera of human heart," *I.E.E.E. Transaction on Medical Imaging*, 10(3), pp295-306.

- [Stanley, 1971] *Introduction to Phases Transitions and Critical Phenomena*, The International Series of Monographs on Physics, Oxford university Press.
- [Szu, 1987] "Nonconvex Optimization," *S.P.I.E.*, 968, pp 59-65.
- [Takeda & Goodman, 1986] "Neural networks for computation : Number Representations and Programming Complexity," *Applied Optics*, 25(18), pp 3033-3046.
- [Terzopoulos, 1983] "Multilevel computational processes for visual surface reconstruction," *Computer Vision, Graphics and Image Processing*, 24, pp 52-96.
- [Terzopoulos, 1984] "Multiresolution computation of visible-surface representations," *Ph.D. Thesis*, Department of Electrical Engineering and Computer Science, M.I.T..
- [Thompson & Kearny, 1987] "Optical flow estimation : an error analysis of gradient-based methods with local optimization," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 9(2), pp 229-244.
- [Tikhonov, 1963] "Solution of incorrectly formulated problems and the regularization method," *Soviet Math. Dokl.*, 4, pp 1035-1038.
- [Tikhonov & Arsenin, 1977] *Solution of ill-posed problems*, Winston & Sons, Washington D.C..
- [Torre & Poggio, 1986] "On edge detection," *I.E.E.E. Transaction on Pattern Analysis and Machine Intelligence*, 8, pp 147-163.
- [Ullmann, 1976] "Filling in the gaps : the shape of subjective contours and a model for their generation," *Biological Cybernetic*, 25, pp 1-6.
- [Verri & Poggio, 1986] Against quantitative optical flow, *memo n°917*, Artificial Intelligence Laboratory, MIT, Cambridge, MA.
- [Wabba, 1980] "Ill-posed problems : numerical and statistical methods for mildly, moderately and severely ill-posed problems with noisy data," *Technical report n° 595*, University of Wisconsin, Madison.
- [Werbos, 1974] "Beyond Regression : New Tools for Prediction and Analysis in the Behavioral Sciences," *Ph.D. dissertation*, Harvard University, Cambridge, MA.
- [Wintz, 1977] *Digital Image Processing*, Addison-Wesley, Reading, MA.

-
- [Widrow & Hoff, 1960] "Adaptative Switching Circuits," *Institute of Radio Engineers, Western Electronic Show and Convention, Convention Record*, 4, pp 96-104, and reprinted in *Neurocomputing*, Anderson, J.A. and Rosenfeld, E. (1988). MIT Press, Cambridge, MA, pp 92-114.
- [Widrow, 1962] "Generalization and information storage in networks of adaline neurons," *Self-Organizing Systems*, pp 435-461, M. Yovitz, G. Jacobi, G. Goldstein, Eds., Spartan Books, Washington DC.
- [Widrow & Lehr, 1990] "30 Years of Adaptive Neural Networks : Perceptron, Madaline, and Backpropagation," *Proceedings of I.E.E.E.*, 78 (9), pp 1415-1440.
- [Yang, 1992] "Mesure du champ de déplacement des matériaux sous charge mécanique par analyse des images," *Thèse de Doctorat*, Ecole Nationale des Ponts et Chaussées.
- [Zhou & Chellappa, 1988] "Image Restoration using a Neural Network," *I.E.E.E. Transaction on Acoustics, Speech, and Signal Processing*, 36, pp 1141-1151.

Publications et Communications

- [Yacoubi, Lelaurin, Devlaminck, & Dubus, 1994a] "Contribution des réseaux neuromimétiques à l'estimation du mouvement des cavités cardiaques". *Actes du Septième Forum Jeunes Chercheurs en Biologique et Médecine*, Lyon, pp. 6-7.
- [Yacoubi, Lelaurin, Devlaminck, & Dubus, 1994b] "Motion Estimation Using a Neural Network". *Proceedings of the International Conference on Electrical & Electronic Engineering (I.C.E.E.E.-94)*, University of Sciences and Technology of Oran, Algeria, 2, pp. 306 - 312.
- [Lelaurin, Yacoubi, Devlaminck, & Dubus, 1994a] "Contribution d'un réseau de neurones à l'estimation des paramètres du mouvement dans une séquence d'images". *Colloque sur le neuromimétisme, A.I.D.R.I.*, Lyon, pp. 271 - 275.
- [Lelaurin, Yacoubi, Devlaminck, & Dubus, 1994b] "A Neural Network for Motion Estimation". *Proceedings of the 2nd Japan - France Congress of Mechatronics*, Takamatsu, Kagawa, Japan, 2, pp. 549 - 552.
- [Devlaminck, Lelaurin, Yacoubi, & Dubus, 1995] "Estimation of Discontinuous Optical Flow Using a Neural Network for Echocardiographic Images". *Medical Imaging 1995, SPIE - Image Processing*, San Diego, California, USA, pp. 210-211.
- [Yacoubi, Lelaurin, Devlaminck, & Dubus, 1995a] "Contribution des réseaux neuromimétiques à l'estimation du mouvement des cavités cardiaques," *Innovation et Technologie en Biologie et Médecine*, 16(4), pp 441-453.

- [Yacoubi, Lelaurin, Devlaminck, & Dubus, 1995b] "Optimization by Neural Networks: Application of Motion Estimation," *I.E.E.E. Engineering in medicine and biology 17th Annal Conference (E.M.B.C.'95) -Signal Processing*, 4, Montréal, Canada, pp 833-834.
- [Lelaurin, Yacoubi, Devlaminck, & Dubus, 1995] "Motion Estimation by using a Neuromimetic Approach," *I.E.E.E. International Conference on Neural Networks (I.C.N.N.'95)*, Perth, Western Australia, 6, pp 2917-2921.
- [Devlaminck, Lelaurin, Yacoubi, & Dubus, 1996] "Stress measure and non rigide motion from density images using a neural network," *Accepté à I.E.E.E. Computation Engeeniring in Systems Applications (CESA'96), IMACS*, Lille.



Résumé

Ce mémoire de thèse traite d'un modèle générique de réseaux de neurones pour la minimisation de fonctionnelles de type quadratique. L'application particulière à l'estimation du mouvement dans des séquences d'images est présentée.

De nombreux problèmes de traitement des images s'expriment sous la forme d'une fonctionnelle à minimiser. Ces derniers sont pour la majorité d'entre-eux mal-posés au sens défini par Hadamard, et nécessitent de recourir à des techniques de régularisation. La résolution de tels problèmes régularisés fait alors appel à des techniques d'optimisation.

Les réseaux de neurones artificiels se composent d'un grand nombre d'unités élémentaires opérant en parallèle; ces unités se désignent sous le terme de *neurones* ou *neurones formels*. De tels systèmes connexionnistes peuvent assurer des tâches de haut niveau telles que la reconnaissance des formes, la classification, l'optimisation,

Nous présentons une *méthodologie neuronale* dont le principe consiste à coder la solution des problèmes d'optimisation en termes d'états de neurone, qui sont les variables discrètes d'un espace euclidien. Le réseau de neurones proposé est fondé sur le modèle discret de Hopfield.

Nous l'appliquons à l'estimation du mouvement dans des séquences d'images, problème qui nous permet de faire une analyse de notre approche la plus complète. Nos résultats sont comparés avec ceux obtenus par l'approche classique proposée par Horn et Schunk. Nous nous étendons ensuite au cas du mouvement d'objets déformables et montrons la nature générique de ce réseau.

Mots clés

Réseaux de neurones, Optimisation, Régularisation, Mouvement, Réseau de Hopfield, Flot optique.