

Le Perceptron multi-couches pour la détection des arêtes d'une image

M. Mkaouar et R. Lepage

Laboratoire d'imagerie, de vision et d'intelligence artificielle

Département de génie électrique

École de technologie supérieure

Montréal, Canada, H2T 2C8

e-mail: mkaouar@ele.etsmtl.ca et lepage@ele.etsmtl.ca

Résumé- Les arêtes fournissent des informations structurelles très compactes sur les objets formant une scène. Elles simplifient ainsi l'analyse de l'image en réduisant la quantité d'information à traiter. Il existe plusieurs techniques d'extraction des caractéristiques de l'image. Nous proposons dans cet article une approche basée sur les réseaux de neurones pour détecter les arêtes de l'image. La tâche de rehaussement des discontinuités est confiée à un premier réseau multi-couches. Un deuxième réseau multi-couches assure la localisation précise des caractéristiques rehaussées et le rejet des détails inutiles comme le bruit ou les pixels à faible module du gradient. Les essais effectués sur un ensemble d'images montrent que l'approche d'extraction des caractéristiques de l'image avec des réseaux de neurones est prometteuse. D'une part, elle ne nécessite aucun seuil pour former le croquis binaire, et d'autre part, elle produit un croquis avec des arêtes d'un seul pixel d'épaisseur. La détection d'arêtes à l'aide des réseaux de neurones permet ainsi d'éliminer d'une façon automatique et efficace le bruit et de localiser les arêtes sans l'intervention de l'utilisateur ni l'utilisation de procédure de traitement secondaire. Le détecteur d'arêtes présenté pourra aussi servir dans une analyse multirésolution de l'image.

I. Introduction

Une arête se manifeste principalement par une variation brusque dans l'illuminance de l'image. Différents phénomènes physiques peuvent se trouver à l'origine des arêtes, comme par exemple le contour d'un objet, la texture, le reflet et/ou un changement d'orientation des surfaces, l'éclairage et le changement de profondeur. L'imbrication de l'effet de ces phénomènes physiques complique l'analyse des scènes réelles qui se présentent souvent entachées de bruit.

Plusieurs chercheurs ont essayé de trouver une solution au problème de détection d'arêtes [1][2][3]. Le

but recherché est de détecter et localiser les caractéristiques significatives de l'image avec la meilleure précision possible.

Il existe, en fait, une multitude d'approches pour l'extraction des caractéristiques de l'image. Néanmoins, détecter les arêtes dans l'image ou encore localiser un objet dans la scène et le reconnaître sont des problèmes dont les solutions sont actuellement incomplètes. Or, nous humains, nous effectuons ces tâches sans même y penser. Devant cette situation, on pense naturellement s'inspirer du système visuel des mammifères pour l'extraction des caractéristiques de l'image. C'est là que réside l'attrait de l'approche d'un détecteur d'arêtes basé sur les réseaux de neurones.

Lepage et Crevier [4] ont développé un réseau de neurones multirésolution pour l'extraction des caractéristiques de l'image. Ils utilisent une structure de donnée pyramidale pour générer les différentes résolutions de l'image. Après avoir rehaussé les discontinuités aux différentes résolutions avec l'opérateur de Sobel, ils utilisent un processus de compétition/coopération pour amincir les arêtes. Simultanément, les arêtes des résolutions grossières renforcent les arêtes correspondantes aux fines résolutions. Ainsi, grâce au processus de compétition/coopération et au contrôle par les couches supérieures de l'information détectée aux couches inférieures, les arêtes parasites tendent à disparaître. Une implantation algorithmique de ce détecteur [12] a été utilisée pour l'apprentissage des réseaux de neurones décrits plus bas.

Moura et Martin [5] utilisent aussi les réseaux de compétition/coopération pour détecter les arêtes. Ils calculent le module ainsi que l'angle du gradient par le biais de l'opérateur de Sobel. Le réseau de neurones utilise ces informations pour renforcer le module des arêtes perpendiculaires au gradient et affaiblir le module des arêtes parallèles au gradient.

À l'aide des réseaux multi-couches, Etemad et Chellapa [6] détectent la présence ou l'absence d'arêtes dans des blocs de taille 3x3 ou 5x5 pixels extraits de l'image d'illuminance. En plus, ils utilisent quelques neurones à sortie linéaire pour estimer le module de l'arête détectée. Cette information permet de rejeter les détails parasites.

Pour détecter les arêtes dans l'image, notre stratégie est la suivante (voir Figure 1): présenter l'image d'illuminance à un premier réseau de neurones pour rehausser les discontinuités. L'image de sortie de cette première opération servira comme entrée pour un second réseau qui cherche à amincir les arêtes, les localiser et éliminer le bruit et les détails à faible amplitude du gradient.

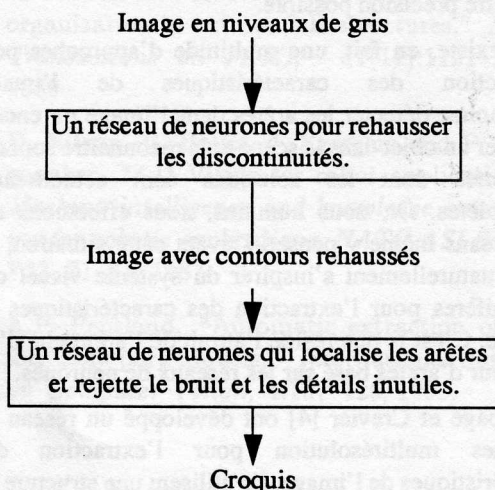


Fig. 1 Extraction des caractéristiques de l'image par des réseaux de neurones.

La section suivante rappelle quelques propriétés des Perceptrons multi-couches. Le propos de la section III est de décrire l'architecture des réseaux de neurones utilisés dans le présent travail. La section IV décrit la construction d'une base d'apprentissage pour le problème particulier de recherche d'arêtes dans une image. Le protocole d'apprentissage ainsi que ses paramètres sont le sujet de la section V. Après l'apprentissage, les capacités du réseau sont évaluées lors d'une phase de généralisation, c'est là le sujet de la section VI qui présente quelques résultats expérimentaux.

II. Le Perceptron multi-couches

Rechercher les arêtes dans l'image consiste à classer les pixels de l'image en deux classes: *contour* ou *non-contour*. Une façon de résoudre ce problème de

classification linéairement non-séparable est d'utiliser des réseaux de neurones multi-couches [7].

Il existe plusieurs variantes qui modifient légèrement l'algorithme de rétro-propagation de l'erreur [8]. L'objectif de cet article est cependant de déterminer les aptitudes des réseaux neuroniques, et plus particulièrement des réseaux multi-couches, à la détection d'arêtes. On a donc utilisé exclusivement l'algorithme de base de rétro-propagation de l'erreur proposé dans le simulateur SNNS (*Stuttgart Neural Network Simulator*) [9] pour effectuer les différentes simulations, en utilisant une valeur nulle pour le momentum.

III. Architecture des réseaux

La couche d'entrée des réseaux considère chaque point de l'image du gradient (ou d'illuminance) avec ses voisins immédiats dans un bloc de taille 5x5 pixels. Ceci permet de rester dans les limites acceptables au niveau de la dimension du réseau. La couche d'entrée contient donc 25 neurones. La couche de sortie ne contient qu'un seul neurone. Lors de l'apprentissage, le tuteur du réseau d'amincissement d'arêtes et d'élimination du bruit étiquette le pixel situé au centre du bloc d'entrée comme *contour* (sortie égale à 0.9) ou *non-contour* (sortie égale à 0.1). Alors que pour le réseau de rehaussement des discontinuités, le tuteur fournit une estimation pour l'amplitude du gradient du pixel situé au centre du bloc d'entrée (valeur comprise entre 0.1 et 0.9).

Aucun pré-traitement n'est appliqué sur les blocs d'entrée. Ainsi, les niveaux de gris de l'image gradient (ou d'illuminance) sont codés directement sur l'intervalle [0,1] (0 correspond à l'illuminance 0, et 1 correspond à l'illuminance 255).

Le nombre de neurones des couches d'entrée et de sortie est ainsi fixé. Cependant, dans un réseau multi-couches, le nombre de couches cachées et le nombre de neurones nécessaire n'est pas prédit par des théorèmes. Un mauvais choix pourrait induire des problèmes comme le surapprentissage ou des délais prohibitifs d'apprentissage.

À ce stade, on souligne deux points importants. Le premier est que le problème de détection d'arêtes est un problème d'appariement entre deux images. Le deuxième est qu'il est possible d'approcher n'importe quelle relation d'appariement avec un réseau multi-couches [10] [11]. Il est donc possible de réaliser un détecteur d'arêtes à l'aide d'un réseau de neurones multi-couches, même si l'architecture exacte n'est pas encore connue.

Pour le réseau de rehaussement des discontinuités, et avec la base d'apprentissage qu'on a choisi, des architectures à une seule couche cachée de 5, 10, ou 16 neurones ont été essayées. D'autres essais avec des réseaux à deux couches cachées de taille (16, 9) et (18, 10) ont aussi été effectués. La réponse varie d'un réseau à un autre. On présente ici la réponse du réseau à deux couches cachées de taille (16, 9), car c'est celui qui a présenté la meilleure réponse lors des tests d'évaluation.

L'apprentissage du réseau d'amincissement d'arêtes et d'élimination du bruit est plus long et plus difficile que celui du réseau de rehaussement des discontinuités. Après plusieurs essais, il s'est avéré impossible de trouver une solution avec un réseau à une seule couche cachée. Des réseaux à deux couches cachées ont été explorés. L'architecture choisie comprend une première couche cachée à 16 neurones, et une seconde à 9 neurones.

IV. Base d'apprentissage

Une base d'apprentissage qui tient compte des spécificités du problème à résoudre offre la possibilité de trouver une solution efficace et complète. En plus, elle permet d'une part d'accélérer l'apprentissage, et d'autre part, de simplifier l'architecture du réseau: réduction du nombre de neurones et du nombre de connexions.

Dans une image, les pixels voisins sont généralement très corrélés et les contours contiennent les pixels les plus riches en information. Les primitives les plus pertinentes se trouvent dans les arêtes d'orientations diverses.

Pour construire une base d'apprentissage, on a considéré l'image de la Figure 2 (a). Cette image est intéressante par la variété des structures rencontrées (voir Figure 2 (c)): homme, fenêtres, des lignes droites, des arcs de cercle et des jonctions en L ou en T). Elle induit de plus des problèmes de suivi et de fermeture de contours qui sont au cœur des difficultés rencontrées lors de l'extraction des caractéristiques de l'image. Pour optimiser la taille de la base d'apprentissage, et avoir en même temps le maximum d'information possible, une distance euclidienne minimum d_{\min} (équation 1) doit exister entre les éléments de la base.

$$d_{ij} = \|X_i - X_j\| = \left[\sum_{n=1}^N (x_{in} - x_{jn})^2 \right]^{1/2} \geq d_{\min} \quad \forall (X_i, X_j) \quad (1)$$

où $X_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{iN})$ et $X_j = (x_{j1}, x_{j2}, x_{j3}, \dots, x_{jN})$ sont deux vecteurs de la base d'apprentissage et $N = 25$.

L'image de la Figure 2 (a) est de taille $108 \times 108 = 11664$ pixels. Une base d'apprentissage avec au maximum 11664 blocs de taille 5×5 pourra donc être

formée. Après l'application d'un détecteur basé sur une pyramide multirésolution [12] sur cette dernière image (on notera ce détecteur PM dans ce qui suit), 1064 pixels ont été déclarés comme *contour* et le reste, soit 10600 pixels, comme *non-contour*. En adoptant un seuil de distance euclidienne de $d_{\min} = 0,185$ entre les vecteurs inclus dans la base, on a 760 parmi les 1064 blocs *contours* qui satisfont à cette condition. On trouve alors que 71,43% des blocs *contours* contiennent des informations différentes les uns des autres. Pour les points *non-contours*, et avec la même distance euclidienne, 2890 blocs ont été choisis. 27,26% seulement des points *non-contours* forment donc un ensemble de vecteurs qui méritent d'être inclus dans la base d'apprentissage. Ce résultat montre que les points *contours* dans l'image sont les plus riches en information et les plus pertinents pour l'apprentissage d'un détecteur d'arêtes qui doit avoir une bonne capacité de généralisation.

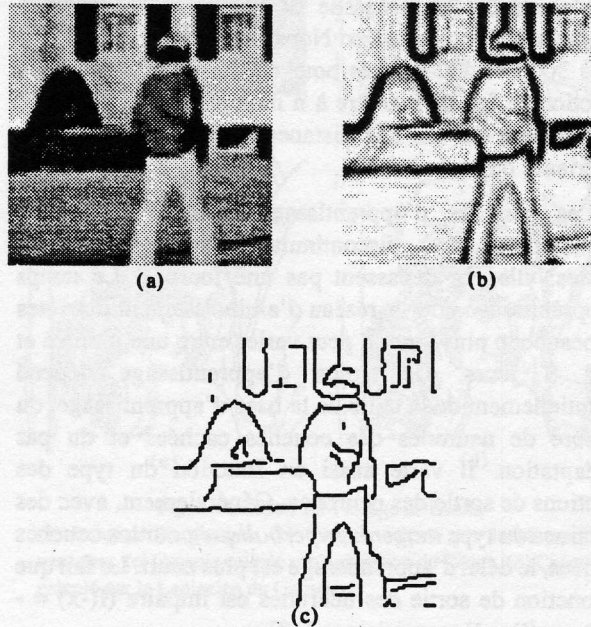


Fig. 2 Images utilisées pour l'apprentissage des réseaux (a) Image d'illuminance. (b) Image avec discontinuités rehaussées avec la dérivée d'une gaussienne de $\sigma = 0,7$. (c) Croquis obtenu à l'aide de l'opérateur PM décrit dans [12] à une résolution $\sigma = 0,7$.

V. Apprentissage

Cette section discute du protocole d'apprentissage, c'est à dire des considérations pratiques qui entourent l'entraînement du réseau.

A. Paramètres d'apprentissage

Pour la phase d'apprentissage, on doit définir un pas d'adaptation η . Or, pour un algorithme du type gradient,

le pas d'adaptation doit être choisi suffisamment grand si l'on cherche une vitesse de convergence acceptable, sans toutefois entraîner de risque d'oscillations de l'algorithme. Actuellement, on ne dispose pas de règles qui prédisent un pas d'adaptation avec une vitesse de convergence satisfaisante. On a alors procédé par essais pour déterminer un pas d'apprentissage acceptable. En fait, c'est avec un pas égal à 0.1 que la majorité des simulations ont été réalisées.

De manière à avoir un bon tuteur pour le réseau d'amincissement d'arêtes et d'élimination du bruit, nous avons détecté les arêtes de façon classique, en utilisant le détecteur PM. Le résultat de cette opération (Figure 2 (c)) est utilisé pour bâtir la base d'apprentissage. Le patron de supervision pour le réseau de rehaussement des discontinuités est obtenu par l'application de la dérivée de la fonction gaussienne sur l'image d'illuminance de la Figure 2 (a). L'image résultante de cette opération (Figure 2 (b)) servira pour former la base d'apprentissage.

L'apprentissage nécessite de présenter des couples (entrée, sortie) aux réseaux. Nous avons donc sélectionné entre 3000 et 4000 paires pour chacun des réseaux. La sélection se fait de manière à n'inclure dans la base que des vecteurs qui ont une distance euclidienne entre eux supérieure ou égale à 0,185.

Les périodes d'apprentissage pour le réseau de rehaussement des discontinuités était relativement courtes: elles ne dépassent pas une journée. Le temps d'apprentissage pour le réseau d'amincissement d'arêtes est beaucoup plus long. Il peut varier entre une journée et 2 à 3 jours. Le temps d'apprentissage dépend essentiellement de la taille de la base d'apprentissage, du nombre de neurones des couches cachées et du pas d'adaptation. Il varie aussi en fonction du type des fonctions de sortie des neurones. Généralement, avec des fonctions du type *tangente hyperbolique* pour les couches cachées, le délai d'apprentissage est plus court. Le fait que la fonction de sortie des neurones est impaire ($f(-x) = -f(x)$) accélère l'apprentissage [13].

La convergence de l'algorithme de rétro-propagation du gradient n'est pas assurée pour un nombre fini et petit d'itérations. En pratique, il faut fixer un indice de convergence. La façon la plus courante est d'arrêter l'apprentissage quand l'erreur quadratique est minimale. Lors des sessions d'apprentissage, un seuil d'arrêt de 0,1 a été considéré. Ce critère ne donne pas d'information sur la qualité de la solution trouvée. Le critère subjectif qui permet de se faire une idée de la qualité de la solution est de la comparer avec le résultat d'un détecteur de contours classique.

Tableau récapitulatif

Nombre de neurones des couches cachées	(16, 9)
Nombre de neurone de la couche de sortie	1
Fonction de sortie des couches cachées	Tangente hyperbolique
Fonction de sortie de la couche de sortie	Logistique
Pas d'apprentissage	0,1
Distance euclidienne entre les patrons	0,185
Résolution de l'image d'apprentissage	$\sigma = 0,7$

B. Évaluation de l'apprentissage

À ce stade, on a donc calculé les poids des réseaux par apprentissage supervisé. Il faut maintenant examiner la réponse des réseaux. Pour ce faire, il faut présenter au premier réseau tous les blocs 5x5 de l'image de la Figure 2 (a), stocker à chaque fois le résultat calculé par le réseau et reconstituer finalement l'image avec arêtes rehaussées. La Figure 3 (b) est le résultat de cette opération. L'image binaire est obtenue d'une façon similaire: on présente l'image aux contours rehaussés au réseau d'amincissement d'arêtes, puis on reconstitue le croquis à partir des sorties du réseau. La Figure 4 montre une comparaison entre l'image contour obtenu par l'opérateur PM et la réponse du réseau de neurones.

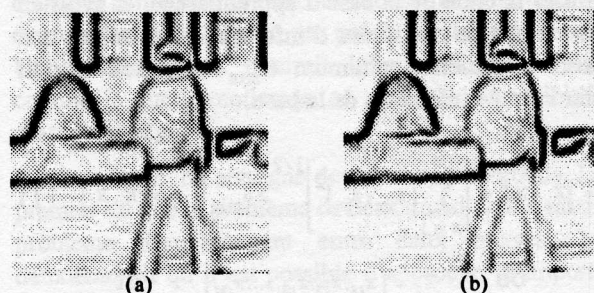


Fig. 3 Réponse du réseau de rehaussement des discontinuités. (a) Image obtenue à l'aide de la dérivée de la fonction de Gauss. Image de taille 108x108 = 11664 pixels. 3500 pixels de cette image ont été choisis pour l'apprentissage. (b) Réponse du réseau à l'image d'illuminance de la Figure 2 (a).

L'image aux arêtes rehaussés de la Figure 3 (b) contient toutes les caractéristiques pertinentes qu'on peut observer dans la Figure 3 (a) qui est obtenue par le biais d'un filtre différentiateur classique. Notons aussi que le fond de la Figure 3 (b) contient moins de petits détails que celui la Figure 3 (a). Ce dernier phénomène peut être observé généralement dans une image qui a été filtrée par un opérateur de lissage. En plus du fait qu'il rehausse les arêtes, le réseau de neurones est donc robuste au bruit.

En comparant le croquis utilisé lors de l'apprentissage (Figure 4 (a)) à la réponse du réseau de neurones (Figure 4 (b)), on remarque que le réseau a détecté des arêtes qui n'ont pas participé à l'apprentissage (par exemple: arêtes détectées sur le dos de l'homme). De plus, de nombreux pixels du fond qui n'ont pas participé à l'apprentissage ont été correctement classés. Ce comportement permet de vérifier que le réseau est capable de généraliser et ne souffre pas du handicap de la surparamétrisation. Ce dernier terme veut dire que le nombre de connexions est trop grand par rapport aux nombre d'exemples dans la base d'apprentissage. Dans ce cas, tous les prototypes seront parfaitement appris. On parle alors d'apprentissage par cœur, mais la réponse du réseau à des exemples différents, même très voisins de ce qui a été appris, risque d'être très mauvaise.

La comparaison entre le croquis du détecteur neuronal et celui obtenu par son maître PM pose une question naturelle: est-ce-que les performances du détecteur neuronal dépassent celles du maître? Bien qu'il est souvent difficile de quantifier avec précision les performances d'un détecteur d'arêtes, un travail futur proposera des mesures de performances dans le but de répondre à une telle question.

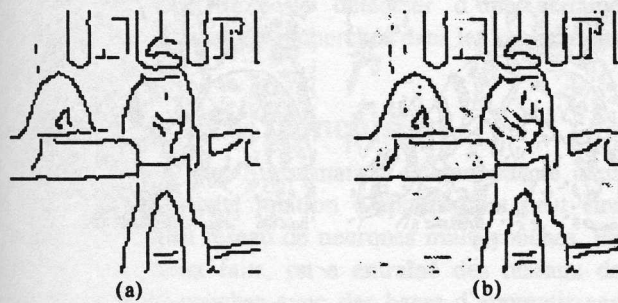


Fig. 4 Réponse du réseau d'amincissement d'arêtes.

(a) Croquis obtenu avec l'opérateur PM. Image de taille 108x108 = 11664pixels. 3650 pixels de cette image ont été choisis pour l'apprentissage. (b) Réponse du réseau à l'image gradient de la Figure 2 (b).

VI. Résultats expérimentaux

Le réseau de neurones a trouvé une fonction d'appariement entre les couples (entrée, sortie) à partir d'exemples qui lui ont été fournis dans la phase d'apprentissage. La phase de test, dite de généralisation, consiste à mesurer les performances de la fonction d'appariement sur des exemples non appris.

Dans cette section, on teste les réseaux sur différentes images. Une comparaison entre ces résultats et des croquis obtenus par l'opérateur de Canny (obtenu du logiciel KBVision) et par le Laplacien de Gaussien est donnée.



Fig. 5 Résultats sur l'image de Lenna

(a) Image d'illuminance. (b) Croquis calculé par le réseau de neurones. (c) Croquis calculé par l'opérateur de Canny. (d) Croquis calculé par le Laplacien du Gaussien.

L'opérateur de détection d'arêtes a été testé dans un premier temps sur deux images largement utilisées dans la littérature. Il s'agit de l'image de Lenna et l'image de la maison (Figure 5 et Figure 6). Dans l'image de Lenna, on note que la plupart des arêtes sont courbées. Cette image sert donc à évaluer la capacité de l'opérateur sur ce type d'arêtes. En plus du croquis obtenu par les réseaux de neurones (Figure 5 (b)), on présente les résultats obtenus par le Laplacien du Gaussien (LoG) ainsi que par l'opérateur de Canny (Figure 5 (c) et (d)). Ces deux opérateurs utilisent des procédures de seuillage pour générer le croquis présenté et rejeter les détails à faible amplitude du gradient. Le résultat du réseau de neurones présenté ici n'a subi aucun

traitement secondaire. Le réseau de neurones fournit automatiquement un croquis contenant uniquement l'information structurelle pertinente de l'image. Ce comportement est dû au fait que le réseau joue le rôle d'un classificateur, une fonction où les réseaux de neurones ont déjà montré leur bonne performance. Il classe les points de l'image en deux classes: *contour* ou *non-contour*. Ainsi, le bruit est classé comme *non-contour* et la structure de l'image émerge d'une façon naturelle.

L'opérateur de Canny est construit de façon à ce que sa réponse à un contour bruité soit unique [1]. En d'autres termes, il ne faut pas que sa réponse contienne trop de maxima locaux proches du maximum local définissant la position du contour. Par conséquent, le croquis produit par cet opérateur ne contient que des contours dont l'épaisseur est d'un seul pixel. À cet égard, la réponse du réseau de neurones est très proche de celle de Canny. La réponse du LoG est cependant moins intéressante.

L'algorithme de Canny utilise la procédure de seuillage par hystérésis pour la poursuite et la fermeture des contours. Des détails inutiles, comme les arêtes au dessous des yeux de Lenna, sont déclarés comme caractéristiques importantes. C'est le choix des seuils supérieurs et inférieurs qui produit ce type de problème. Avec le réseau de neurones, on ne trouve pas ce type de détails dans le croquis puisqu'on n'utilise pas d'opération de seuillage.

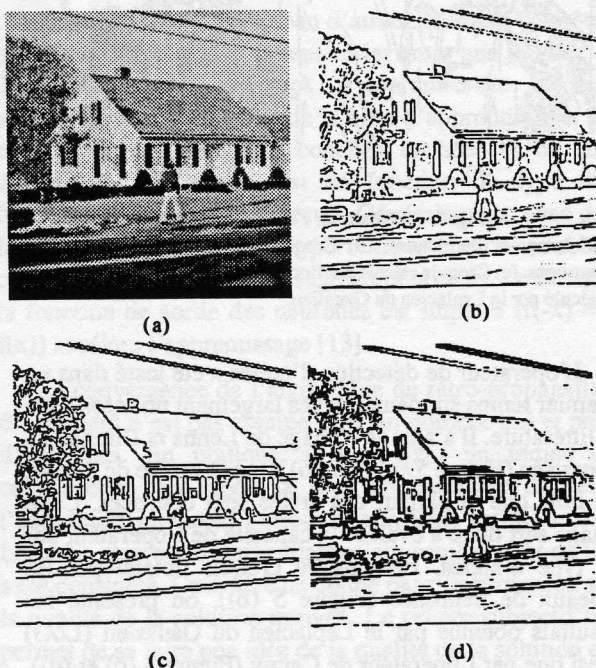


Fig. 6 Résultats sur l'image de la maison

(a) Image d'illuminance. (b) Croquis calculé par le réseau de neurones. (c) Croquis calculé par l'opérateur de Canny. (d) Croquis calculé par le Laplacien du Gaussien.

Le second exemple de test est celui de la maison (Figure 6). L'intérêt de cet image est qu'elle permet d'étudier la capacité des réseaux de neurones dans la détection des lignes droites. En examinant la Figure 6 (b), on trouve des arêtes horizontales et verticales bien tracées ainsi que des jonctions entre les lignes bien formées. L'image de la maison contient aussi un arbre. Notons que le réseau de neurones détecte moins de bruit, au niveau de l'arbre, par rapport à ce que Canny ou le LoG détecte.

Le troisième exemple représente l'image d'une salamandre (Figure 7). Cette image est relativement difficile à traiter. Elle contient des arêtes courbées et beaucoup de petits détails. Il est difficile d'extraire l'animal de la scène sans détecter les détails du fond de l'image. L'opérateur de Canny rencontre lui aussi la même difficulté. Il n'arrive pas à dissocier correctement l'objet du fond de l'image. Le fond du croquis des réseaux de neurones présente des nuages de points, alors que celui de Canny présente des petits détails connectés entre eux. Ceci est dû au phénomène de poursuite des contours utilisé par Canny.

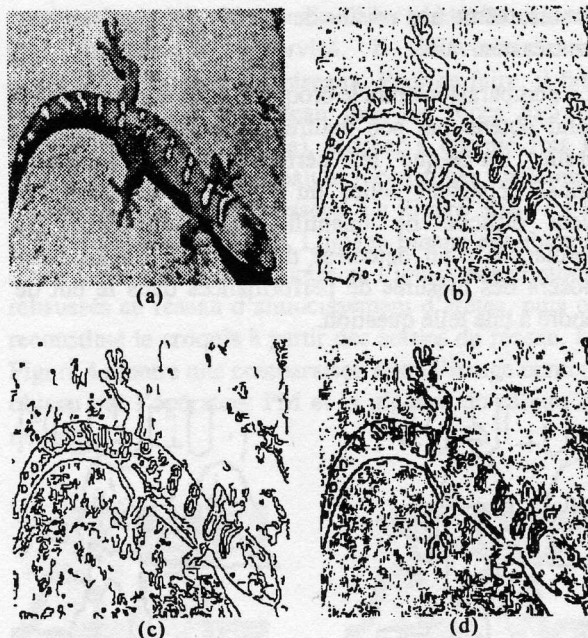


Fig. 7 Résultats sur l'image d'une salamandre

(a) Image d'illuminance. (b) Croquis calculé par le réseau de neurones. (c) Croquis calculé par l'opérateur de Canny. (d) Croquis calculé par le Laplacien du Gaussien.

Une façon efficace d'extraire les caractéristiques de ce type d'image est d'utiliser l'analyse multirésolution. La Figure 8 (a) est le résultat de l'opérateur PM à une résolution $\sigma = 1,5$. La réalisation avec des réseaux de neurones d'un opérateur multirésolution constitue un

objectif futur. La Figure 8 (b) montre un croquis à une résolution 1,5 obtenu par réseau de neurones. Ce réseau a été entraîné à nouveau particulièrement pour fournir des croquis à une telle résolution. L'étape future consiste donc à intégrer les croquis des différentes résolutions.

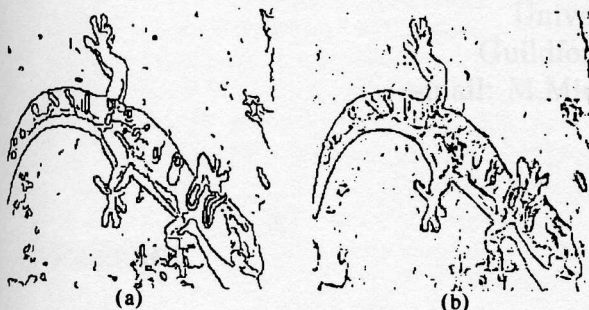


Fig. 8 Analyse multirésolution de l'image par réseau de neurones (a) Croquis obtenu par l'opérateur décrit dans [12] à une résolution $\sigma = 1,5$. (b) Résultat d'un réseau de neurones de résolution 1,5.

Le détecteur présenté dans cet article présente trois caractéristiques importantes. La première est qu'il n'a besoin d'aucun seuil pour classer un point de l'image comme *contour* ou *non-contour*. Cette première caractéristique est très importante puisque le choix du seuil est critique lors de l'extraction des caractéristiques de l'image. La deuxième caractéristique est que le réseau a été entraîné avec une base de donnée de taille limitée et il est utilisé pour extraire les caractéristiques de différents types d'images. Les résultats sont souvent satisfaisants, et se comparent bien à ceux obtenus par les opérateurs classiques. La troisième caractéristique est que la réponse de l'opérateur à une arête quelconque est généralement unique. En fait, il calcule des arêtes d'un seul pixel d'épaisseur. Ceci dote le détecteur d'une certaine précision qui est souvent recherchée dans les applications industrielles.

VII. Conclusion

Il est connu que l'information est redondante dans l'image et que toute relation d'appariement peut être approchée par un réseau de neurones multi-couches. En utilisant ces deux faits, on a entraîné des réseaux de neurones multi-couches avec des bases d'apprentissage de taille limitée pour extraire les caractéristiques de différentes images.

Les réseaux opèrent sur des fenêtres de taille 5x5 pixels et décident si le pixel se trouvant au centre de la fenêtre appartient à une ligne de crête ou non. Grâce à cette procédure de classification, une élimination du bruit dans l'image est obtenue implicitement. Ceci permet

d'obtenir un croquis binaire sans passer par des opérations de seuillage classique.

Après l'apprentissage, les réseaux de neurones passent à l'épreuve d'évaluation de leur capacité de généralisation sur des images réelles. Il est donc naturel d'extraire les échantillons d'apprentissage à partir d'images réelles. Idéalement, les variations d'illuminance dans cette image devraient couvrir tous les schémas de variations d'illuminance rencontrés dans les blocs tests. Ceci est impossible dans la pratique. On se contente alors des exemples les plus représentatifs.

Les résultats expérimentaux montrent que l'extraction des caractéristiques de l'image peut se faire avec des réseaux de neurones multi-couches sans l'utilisation de procédures de traitement d'image classique ni l'intervention de l'utilisateur. Les performances et l'architecture de ce type d'opérateur sont largement liées à la base d'apprentissage. En fait, la base d'apprentissage optimale doit satisfaire deux exigences contradictoires: un minimum d'exemples d'apprentissage et un maximum d'échantillons représentatifs de chaque classe. La première exigence permet de simplifier l'architecture du réseau et d'accélérer l'apprentissage. La deuxième augmente les performances du réseau en généralisation.

En choisissant de n'inclure dans la base d'apprentissage que des vecteurs séparés les uns des autres par une certaine distance euclidienne, on est assuré d'avoir des exemples d'apprentissage riches en information. Le fait d'augmenter la distance euclidienne nous approche davantage d'un ensemble de vecteurs orthogonaux. Un travail futur qui explorerait ce type de base d'apprentissage pourrait peut-être augmenter les performances de cet opérateur.

VIII. Références

- [1] J. Canny, "A Computational Approach to Edge Detection", *IEEE Trans. Pattern Anal. & Mach. Intell.*, Vol. PAMI-8(6), pp. 679-698, Nov. 1986.
- [2] F. Bergholm, "Edge Focusing", *IEEE Trans. Pattern Anal. & Mach. Intell.*, Vol. PAMI-9(6), pp. 726-741, Nov. 1987.
- [3] D. Marr et E. Hildreth, "Theory of Edge Detection", *Proc. R. Soc. London Ser. B*, Vol. 207, pp. 187-217, 1980.
- [4] R. Lepage and D. Crevier, "Multiresolution Neural Network for the Extraction of the Primal Sketch", *Proc. SPIE's Intelligent Robot and Computer Vision XIII: Algorithms and Computer Vision*, Vol. 2355, 31 October - 2 November, Boston, Massachusetts, David P. Casasent, Ed, pp. 375-384, SPIE, 1994.
- [5] L. Moura et F. Martin, "Edge Detection Through

- Cooperation and Competition”, *Proc. IJCNN 1991, Singapore, Vol. 3*, pp. 2588-2593.
- [6] K. Etemad et R. Chellappa, “A Neural Network Based Edge Detector”, *1993 IEEE International Conference on Neural Network*, Vol. 1, March 28 - April 1, San Francisco, pp. 132-137.
- [7] D.E. Rumelhart, G.E. Hinton et R.J. Williams, “Learning Internal Representations by Error Propagation”, in D.E. Rumelhart et J.L. McClelland, *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*, Vol 1, Ch. 8., pp. 318-362, Cambridge, Mass., MIT Press., 1986.
- [8] R. Battiti, “First and Second-Order Methods for Learning”, *Neural Computation*, 4, pp. 141-166, 1992.
- [9] A. Zell, N. Mache, R. Hübner, G. Mamier, M. Vogt, K. Herrmann, M. Schmalzl, T. Sommer, A. Hatzigeorgiou, S. Döring, D. Posselt, *SNNS, Stuttgart Neural Network Simulator*, 244p., University of Stuttgart, 1993.
- [10] K. I. Funahshi, “On the Approximate Realization of Continuous Mappings by Neural Networks”, *Neural Networks*, Vol. 2, pp. 183-192, 1989.
- [11] K. Hornik, M. Stinchcombe et H. White, “Multilayer Feedforward Networks are Universal Approximators”, *Neural Networks*, Vol. 2, pp. 359-366, 1989.
- [12] M. Mkaouar et R. Lepage, “Extraction des caractéristiques d’une image par analyse à multiple résolutions spatiales”, *Congrès canadien de génie électrique et informatique 1995*, Vol.2, pp. 1176-1179, Montréal 5-8 septembre.
- [13] S. Haykin, *Neural Networks, A Comprehensive Foundation*, Macmillan College Publishing Company, IEEE Press, 1994.