

G

Département de géomatique appliquée
Faculté des lettres et sciences humaines
Université de Sherbrooke

G
70.4
C535
2011

Représentation adaptative d'images de télédétection à très haute
résolution spatiale : une nouvelle approche hybride
(la décomposition pyramidale avec des réseaux de neurones)

Valeriy Cherkashyn

**Thèse présentée pour l'obtention du grade de Philosophiae Doctor (Ph. D.)
en télédétection**

Février 2011

© Valeriy Cherkashyn, 2011

I-2552

Composition du jury

Cette thèse a été évaluée par un jury composé des personnes suivantes :

Ph.D. Dong-Chen He, directeur de recherche
(Département de géomatique appliquée, FLSH, CARTEL, Université de Sherbrooke)

Ph.D. Goze Bertin Béné, examinateur interne
(Département de géomatique appliquée, FLSH, CARTEL, Université de Sherbrooke)

Ph.D. Samuel Foucher, examinateur interne
(Centre de recherche informatique de Montréal, CRIM)

Ph.D. Wei Li, examinateur externe
(Centre de recherches sur les communications Canada, CRC)

Résumé

Valeriy Cherkashyn (2010) Représentation adaptative d'images de télédétection à très haute résolution spatiale: une nouvelle approche hybride (la décomposition pyramidale avec des réseaux de neurones). Thèse présentée pour l'obtention du grade de Philosophiae Doctor (Ph.D.) en télédétection. Département de géomatique appliquée, CARTEL, Université de Sherbrooke, Sherbrooke (Québec), Canada, 116 p.

De nos jours l'observation de la terre à l'aide d'images satellitaires de très haute résolution spatiale (Ikonos, Quickbird, WorldView-2) donne de nombreuses possibilités pour gérer de l'information à l'échelle mondiale. Les technologies actuelles d'acquisition d'information sont à l'origine de l'augmentation importante du volume des données.

L'objectif général de cette thèse consiste à développer une nouvelle méthode hybride de représentation d'image numérique de très haute résolution spatiale qui améliore la qualité visuelle d'images compressée avec un haut niveau de compression (100 fois et plus).

La nouvelle méthode hybride exploite la transformation pyramidale inverse d'image numérique en utilisant des réseaux de neurones artificiels. Elle combine le traitement spatial et la transformation abstraite de l'image.

L'emploi de l'approche de la transformation pyramidale inverse a démontré l'efficacité du traitement de l'information à une ou à des échelles spécifiques, sans interférer ou ajouter un temps de calcul inutile. Cette approche est essentielle pour réaliser une transformation progressive d'image. Les résultats montrent une amélioration du rapport signal pur bruit de 4 dB pour chaque couche additionnelle de la transformation progressive.

Nous avons réussi à garder une qualité visuelle d'images compressées comparable, jusqu'au niveau de la compression de 107 fois. De plus, pour le niveau de la compression de 274 fois, nous avons obtenu une amélioration de la qualité visuelle en comparaison des méthodes de compression courantes (JPEG, JPEG2000).

Les résultats du travail confirment l'hypothèse que les images de télédétection possèdent un haut degré de redondance et que l'utilisation d'un réseau de neurones est un bon moyen pour trouver l'opérateur efficace du regroupement de pixels.

Cette nouvelle méthode de représentation d'images à très haute résolution spatiale permet de réduire le volume des données sans détérioration majeure de la qualité visuelle, comparé aux méthodes existantes.

Enfin, nous recommandons de poursuivre l'exploration du domaine des calculs distribués tels que les réseaux des neurones artificiels, considérant l'augmentation de la performance des outils informatiques (nanotechnologies et calculs parallèles).

Mots-clés : représentation pyramidale inverse, réseau de neurones artificiels, télédétection, images satellitaires, système d'information géographique, vision de l'être humain, compression d'images.

Abstract

Valeriy Cherkashyn (2010) Adaptive representation of satellite images very high definition: a new hybrid approach (pyramidal inverse decomposition with neural networks). This thesis presented for obtainment of rank of Philosophiae Doctor (Ph.D.) in Remote Sensing (Image Processing). Département de géomatique appliquée, Université de Sherbrooke, Sherbrooke (Québec), Canada, 116 p.

Earth observations using very high-resolution satellite imagery, such as from Ikonos, QuickBird or WorldView-2, provide many possibilities for addressing issues on a global scale. However, the acquisition of high-resolution imagery using these technologies also significantly increases the volume of data that must be managed. With the passing of each day, the number of collected satellite images continues to increase.

The overall objective of this work is to develop new hybrid methods for numerical data representation that improve the visual quality of compressed satellite visible imagery for compression levels of 100 times and more. Our new method exploits the inverse pyramid transform using artificial neural networks, and thus addresses the trend in the field of remote sensing and image compression towards combining the spatial processing and abstract transformation of an image.

Our implementation of the pyramidal inverse transformation demonstrates the effectiveness of information processing for specific levels, without interfering or adding unnecessary computation time. This approach is essential in order to achieve a gradual transformation of an image. The results showed an improvement in the signal to noise ratio of 4dB for each additional layer in the pyramidal image transformation.

We managed to keep a similar level of visual quality for the compressed images up to a compression level of 107 times. In addition, for a compression level of 274, we improved the visual quality as compared to standard compression methods (JPEG, JPEG2000).

The results of this study confirm the hypothesis that remote sensing images have a high degree of redundancy and that the use of neural networks is a good way to find the effective operator of the pixel combination.

This new method for image representation reduces the volume of data without major deterioration in the visual quality of the compressed images, as compared to existing methods.

Finally, we recommend further exploration in the field of distributed computing, such as artificial neural networks, considering the rapidly increasing performance of computers in the near future (parallel computing technology and nanotechnology).

Key words: pyramidal inverse decomposition, artificial neural networks, remote sensing, satellite images, geographical information systems, human visual system, image compression.

À mes ancêtres

Remerciements

Je remercie d'abord Pr Dong-Chen He, mon directeur de recherche, pour l'intérêt qu'il a porté à mon travail, pour son soutien moral, sa patience et sa franchise, ainsi que pour sa vision du problème de recherche et des nouvelles conditions en télédétection depuis le tout début. Je le remercie aussi pour ses conseils pendant nos conversations, qui m'ont permis de réaliser ce travail remarquable et très intéressant et, finalement, pour toute la confiance et la certitude qu'il a eue envers moi et mon projet.

Je tiens aussi à remercier ma femme Lidiya Cherkashyna et notre fille Mariya pour le soutien quotidien et pour l'admirable compréhension du fait que j'ai, pendant un temps, offert moins d'attention à ma famille. J'adresse une reconnaissance extraordinaire à ma mère Nina Cherkashyna pour sa foi en moi, ses prières et son amour. C'est grâce à elle, si j'ai pu être traversé par l'espoir dans certains moments plus difficiles.

Je remercie le Ministère de l'Éducation, du Loisir et du Sport pour l'aide financière aux études (programme de prêts et bourses) qui m'a permis de commencer l'enseignement au niveau supérieur à l'Université de Sherbrooke.

Je remercie aussi la Faculté des Lettres et Sciences humaines, le Département de Géomatique appliquée, le laboratoire du CARTEL et la fondation FORCE pour les différentes bourses et l'aide financière qu'ils m'ont permis d'obtenir pour finir ma thèse dans un délai raisonnable.

Enfin, je remercie le Conseil de Recherche en Sciences naturelles et en Génie du Canada pour la semi-couverture de mes dépenses lors de la présentation des résultats de mes recherches en conférences, ainsi que les publications des articles dans les journaux reconnus. Sans ce soutien, ma thèse n'aurait pas pu venir au monde dans un aussi court laps de temps.

Je voudrais dire merci beaucoup à tous les professeurs, à mes amis, aux étudiants et à tout le personnel de la Faculté des Lettres et des Sciences humaines pour l'accueil chaleureux pendant ce temps. Trouvez ici l'expression de ma profonde reconnaissance.

Table des matières

Composition du jury	i
Résumé	ii
Abstract	iii
Remerciements	iv
Liste des figures	vi
Liste des tableaux	vii
Liste des annexes	vii
Liste des acronymes	vii
1. Introduction	1
1.1. Mise en contexte	1
1.1.1. Mondialisation : les TI et l'information géographique	1
1.1.2. Mondialisation : des images satellitaires comme source d'information géographique indispensable	2
1.2. Problématique de la recherche	3
1.2.1. Pourquoi compresser les données ?	5
1.2.2. Qu'est-ce que la compression de données ?	5
1.2.3. Deux problématiques majeures en recherche	6
1.3. Objectifs	7
1.4. Hypothèses de la recherche	7
1.5. Plan de la thèse	8
2. Revue de littérature	9
2.1. Mise en contexte	9
2.1.1. Contenu informatif des sources de données	9
2.1.2. Taux de compression	10
2.1.3. Paramètre PSNR	11
2.1.4. Similarité structurelle (SSIM)	12
2.2. Traitement d'images et vision artificielle	13
2.3. Deux approches potentielles d'utilisation de l'image de télédétection	13
2.4. Rappel des techniques de représentation	14
2.5. Compression d'image	14
2.5.1. Compression sans perte (les méthodes réversibles)	15
2.5.2. Compression avec perte (les méthodes irréversibles)	16
2.6. Algorithmes de compression actuels	17
2.7. Architecture des algorithmes de compression	20
2.8. Généralités sur les méthodes de représentation d'images	20
2.8.1. Transformation pyramidale (ou décomposition multiéchelles)	20
2.8.2. Traitement de données avec du réseau de neurones artificiels	22
2.8.3. Discussion des méthodes des références	23
3. Méthodologie	24
3.1. Représentation pyramidale inverse différentielle	24
3.1.1. Décomposition d'images par la PID	26
3.1.2. Codage de coefficients de la PID	29
3.1.3. Décodage de données compressées par la PID	30
3.2. Réseau de neurones multicouche	32
3.3. Apprentissage d'un réseau multicouche	34
3.4. Étapes importantes de la méthode proposée	35

3.4.1. Entraînement du réseau de neurones	36
3.4.2. Codage	37
3.4.3. Décodage	37
3.5. Réalisation algorithmique de l'approche hybride proposée	38
3.6. Données des images utilisées	40
4. Résultats	41
4.1. Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "Image decomposition on the basis of an inverse pyramid with 3-layer neural networks", <i>Journal of Communication and Computer</i> , ISSN 1548-7709, USA, Nov. 2009, Vol. 6, No.11 (Serial No.60), pp. 21-29	45
4.2. Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "Compression of high-resolution satellite visible images with pyramidal neural network", International Science Press, <i>International Journal of Neural Networks and Applications</i> , 2(1) January-June 2009, Vol. 2, No 1, pp.5-14	54
4.3. Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "A Novel Adaptive Representation Method (AIPR/BPNN) of Satellite Visible Images Very High Definition", <i>Journal of Communication and Computer</i> , ISSN 1548-7709, USA, Sep. 2010, Vol. 7., No.9 (Serial No.70), pp. 55-66	65
5. Discussion	78
5.1. Formats actuels sur le marché géospatial	78
5.2. Taille du bloc du traitement en fonction de la taille de l'image traitée	78
5.3. Architecture du réseau de neurones	80
5.4. Discussion sur des résultats de paramètre SSIM	81
5.5. Temps des calculs	81
5.6. Correspondance au système d'être humain	82
6. Contribution	83
6.1. Contribution scientifique	83
6.2. Contribution scientifique appliquée	83
7. Originalité du projet	84
8. Conclusions générales	85
8.1. Avantages	86
8.2. Limitations	86
9. Perspective	87
9.1. Nanotechnologie et calculs parallèles	87
9.2. Canal de transmission des données numériques	88
10. Références	89
Annexes	93
Glossaire	105

Liste des figures

Figure 1.1 Processus de la transmission de données par le canal de télécommunication.	5
Figure 2.1 Processus global de la compression – décompression d'images	20
Figure 2.2 Correspondance des niveaux de la pyramide vers des objets que l'on peut définir	21
Figure 2.3 Représentation des données numériques par la méthode classique de la pyramide	21
Figure 2.4 modèle du perceptron. Source : Hu et Hwang, 2002	22

Figure 3.1 Organigramme de la démarche scientifique	25
Figure 3.2 Décomposition pyramidale d'images : le niveau plus bas de la pyramide ($P=2$) et le niveau plus haut de la pyramide ($P=0$)	26
Figure 3.3 Couches $p = 0, p = 1$ de la PID pour l'image $[B_{i,j}]$ de taille $H \times V$ pixels : a. Division de l'image en K sous images de taille $2^n \times 2^n$ pixels dans la couche $p=0$; b. Division de chaque sous-image de la couche $p=0$ (dans les cercles) à quatre sous-images de taille $2^{n-1} \times 2^{n-1}$ pixels dans la couche $p=1$	27
Figure 3.4 Décomposition d'image par la Pyramide Inverse Différentielle (PID) ...	31
Figure 3.5 Réseau de neurones multicouche	32
Figure 3.6 Étape d'apprentissage du réseau de neurones	34
Figure 3.7 Processus d'apprentissage du BPNN	35
Figure 3.8 Étapes de la méthode proposée	36
Figure 3.9 Méthode de la décomposition pyramidale pour une image $m \times m$ par des réseaux de neurones et des méthodes d'entropie du codage de l'information (CED – le codage d'entropie direct ; CEI – le codage d'entropie inverse)	39

Liste des tableaux

Tableau 2.1 Structure pyramidale de 10 niveaux du réseau de neurones à trois couches pour l'image de résolution 512×512 pixels	33
Tableau 5.1 Structure pyramidale de l'image d'entrée (512×512 , 262 144 Octets) en fonction des paramètres du réseau de neurones à trois couches	79
Tableau 5.2 Résultats de choix de la taille de sous-image	80

Liste des annexes

Annexe 1. Caractéristiques principales du radiomètre équipant le satellite Ikonos ..	93
Annexe 2. Étapes de traitement d'images numériques	95
Annexe 3. Vision artificielle et traitement d'images	96
Annexe 4. Système de la vision de l'être humain	97
Annexe 5. Réseau de neurones artificiels	100
Annexe 6. Stéréo satellite image	104

Liste des abréviations, des sigles et des acronymes utilisés

APC	–	Analyse en composantes principales
BPNN	–	Réseau de neurones à rétropropagation (Back Propagation Neural Network)
BTC	–	Encodage par pavé (Block Truncation Coding)
CA	–	Codage arithmétique
CE	–	Codage entropique
CRIM	–	Centre de Recherche informatique de Montréal
CR	–	Taux de compression (Compression rate)
DCT	–	Transformée en cosinus discrète (Discrete Cosine Transform)
DFT	–	Transformée de Fourier discrète (Discrete Fourier Transform)
DWT	–	Transformée en ondelettes discrète (Discrete Wavelet Transform)
ECW	–	Enhanced Compression Wavelets

ERS	–	European Remote-Sensing Satellite
EZW	–	Embedded zero-tree transformée en ondelettes (Embedded zero-tree Wavelet Transform)
FT	–	transformée en fractales (Fractal Transform)
GIF	–	Graphics interchange format
GP	–	Décompositions pyramidale de Gaussien (Gauss Pyramide)
HVS	–	Système de perception visuelle humaine (Human Vision System)
IDP	–	Décomposition à la pyramide inverse (Inverse Pyramidal Decomposition)
FFT	–	Transformée rapide de Fourier (Fast Fourier Transform)
JPEG	–	Joint Photographic Expert Group
KLT	–	Transformée de Karhunen-Loève (Karhunen-Loeve Transform)
LP	–	Transformée basée sur la prédiction linéaire (Linear Prediction)
LZW	–	Lempel-Ziv-Welch algorithme
MLP	–	Architecture multicouche du RN (Multi Layers Perceptron)
MrSid	–	MultiResolution Seamless Image Database
MSE	–	Erreur de la moyenne carrée minimale (l'erreur quadratique moyenne)
PACS	–	Picture Archival and Communication Systems
PID	–	Décomposition par la Pyramide inverse différentielle
PIT	–	Transmission progressive d'image (Progressive Image Transmission)
PL	–	Décompositions pyramidale de Laplace (Laplace pyramide)
PSNR	–	Power signal-to-noise ratio (Peak signal-to-noise ratio)
RBFN	–	Radial Basis Function Neural Network
RLE	–	Codage par plage (Run-Length Encoding)
RN	–	Le réseau de neurones (Neural Network)
SNR	–	Rapport signal sur bruit (Signal-to-noise ratio)
SSIM	–	Mesure de similarité entre deux images numériques (Structural Similarity)
SVD	–	Décomposition en valeurs singulières (Singular Value Decomposition)
THRS	–	Très haute résolution spatiale
TIFF	–	Tagged Image File Format
VATI	–	Vision artificielle et traitement d'images
VLC	–	Encodage à longueur variable (Variable-Length Coding)
VQ	–	Quantification vectorielle (Vector Quantization)
SPOT	–	Satellite pour l'Observation de la Terre
SVD	–	Décomposition en valeurs singulières (Singular Value Decomposition)
WHT	–	Transformée Walsh-Hadamard (Walsh-Hadamard Transform)
a	–	image originale
\hat{a}	–	image compressée
i et j	–	les coordonnées des pixels
Variance	–	variance entre classes
x	–	fenêtre de l'image originale
y	–	fenêtre de l'image compressée

*« La vitesse des communications ne peut pas être augmentée sans
l'aide des techniques de compression. »*

Guillois, J.-P.

1. Introduction

1.1. Mise en contexte

La mondialisation amène un lot de changements dans tous les domaines. L'un des indicateurs de la mondialisation est le développement des technologies de l'information (TI). Le rythme inédit du développement et de la diffusion des TI a un potentiel immense pour propager la mondialisation à l'échelle planétaire (Caloz et Collet, 2001 ; Campbell, 2002). Autrement dit, l'impact des nouvelles TI confère un caractère nouveau et distinctif au processus actuel de mondialisation. Donc, la mondialisation des TI est devenue un outil indispensable de l'intégration sociale et économique. En effet, les TI ont commencé à pénétrer les pays en voie de développement, modifiant ainsi les pratiques de gouvernance du milieu dans différents secteurs.

1.1.1. Mondialisation : les TI et l'information géographique

Au cours des dernières décennies, l'information géographique a connu d'importants développements qui ont permis la création d'applications très intéressantes dans divers domaines comme l'urbanisme, l'archéologie, l'agriculture et la météorologie. L'information géographique est un élément-clé pour la prise de décisions dans de nombreux domaines d'études. En effet, l'information géographique est vitale pour un système d'information géographique (SIG), essentiellement dans les étapes d'acquisition, de traitement, d'analyse et de manipulation de données spatiales.

Le SIG joue un rôle indispensable pour l'analyse et la gestion de toutes sortes de données géographiques (des images numériques, des cartes, des données socioéconomiques, etc.). La donnée géographique présente dans un SIG provient de différentes sources (Steinberg, 2000). En plus, des projets actuels exigent de travailler dans des domaines multidisciplinaires (environnement global, évaluation des risques, médecine, etc.).

1.1.2. Mondialisation : des images satellitaires comme source d'information géographique indispensable

L'information géographique joue aujourd'hui un rôle primordial dans la gestion des ressources naturelles et dans la protection de l'environnement, à l'échelle mondiale. Le développement technologique a permis l'essor des techniques d'acquisition et de traitement de l'information depuis l'apparition des satellites d'observation de la terre, qui eux, ont donné naissance à la télédétection spatiale (Desjardins, 2000 ; Jensen, 2005). La découverte des images de la terre à partir des satellites fait dorénavant partie du quotidien, notamment avec les images télédiffusées lors des bulletins météorologiques.

De nos jours, l'observation de la terre à l'aide d'images satellitaires de très haute résolution (THRS) donne de nombreuses possibilités pour gérer le milieu physique et humain à l'échelle de la planète. Utilisées à des fins opérationnelles, les images satellites de THRS, telles que celles fournies par ERS-SAR, RADARSAT, Ikonos et WorldView-2, offrent de nouveaux débouchés dans les domaines de l'étude et du suivi des ressources naturelles.

La tendance actuelle est de mettre au point des capteurs ayant une résolution spatiale de plus en plus fine. À titre d'exemple, Ikonos est le premier capteur civil de THRS (limite de résolution de 1 m) fournit des images comparables aux photos aériennes. On peut donc faire la différence entre une voiture et un camion.

De nouvelles générations de capteurs tels que WorldView-2 offrent maintenant des images ayant une résolution spatiale de moins de 1 mètre (limite de résolution de 0,41 m).

Remarquons que pour des raisons évidentes de transmission de données, la télédétection satellitaire ne pouvait pas se contenter de fournir des données analogiques (c.-à-d. comme les photographies). Il est évident que la télédétection satellitaire de THRS exige plus d'espace dans les bases de données, créant ainsi de sérieux problèmes lors de la transmission des données volumineuses.

Nous constatons un nombre toujours plus important de services interactifs de distribution d'images de télédétection sur Internet (Pointet *et al.*, 2003). Les besoins actuels et le temps d'en matière de partage de l'information numérique et d'accès à celle-ci croît continuellement. Il faut diminuer cette durée en introduisant de nouvelles méthodes adaptatives de représentation des images.

1.2. Problématique de la recherche

Les technologies actuelles d'acquisition d'information de télédétection sont à l'origine de l'augmentation importante des volumes de données à gérer, stocker, traiter et transférer. Les images de télédétection atteignent des résolutions spatiales de l'ordre de 10 m et peuvent atteindre un mètre, voire un demi-mètre, pour les images satellitaires de THRS (Ikonos, QuickBird et WorldView-2) pour un nombre de bandes spectrales toujours croissant (Tonye, 2000).

En effet, l'accroissement du nombre de bandes et la mise sur le marché d'images de THRS ont augmenté considérablement le volume moyen des images de télédétection. Il n'est plus rare de devoir manipuler des fichiers de plusieurs Gigabits (Gb) pour une seule image satellitaire. La nécessité de réduire la taille des images apparaît indispensable (Caloz et Collet, 2001).

À titre d'exemple, pour une scène Ikonos (voir annexe 1), on compte sur une seule image : $11\ 000\ \text{pixels} * 11\ 000\ \text{pixels} * 1\ \text{bande (noir et blanc)} * 8\ \text{bits par pixel (bbp)}$ soit 121 mégaoctets (Mo) (voir annexe 1). En mouvement, à la vitesse de 7 km par seconde, Ikonos recueille les données en noir et blanc et les données multispectrales avec la performance de 2000 kilomètres carrés par minute. Donc, pour une seule minute de travail d'Ikonos, il faut stocker 2 Gb de données (GeoEye, 2006).

Ikonos est capable de repasser sur la même zone tous les 3 à 5 jours. Pour couvrir le territoire ciblé, il faut faire plusieurs recouvrements. Par exemple, pour l'image couvrant tout le Québec avec un territoire de $1\ 365\ 128\ \text{km}^2$ (avec un paquet de 4 bandes, 11 bits par pixel, avec la résolution de 1 mètre pour la bande panchromatique et 4 mètres pour la bande multispectrale), le volume de données correspond à plus de 3,4 téraoctets (To), soit une pile de CD-ROM de plus de 65 m de hauteur.

Le nombre d'images satellitaires collectées augmente avec le nombre de jours. Ainsi, au bout d'une année, on obtient plusieurs milliers d'images et les fichiers où celles-ci sont archivées deviennent de plus en plus volumineux. Par exemple, pour le radiomètre équipant le capteur Ikonos, les images couvrant 275 millions de kilomètres carrés ont été reçues jusqu'à l'année 2008.

Après leur exploitation, de telles images ne sont plus conservées dans une même place et doivent alors être transformées avant d'être archivées pour de longues périodes. Les questions d'archivage et de stockage des images de THRS s'imposent de ce point de vue. Pour économiser les ressources de stockage ou des canaux de communication, on cherche à compresser le signal numérisé (Barlaud et Labit, 2002).

L'augmentation du volume des images produit aussi de sérieux problèmes de gestion et de partage de l'information. Le transfert de l'information dans les bases des données (Steinberg, 2000) rencontre des difficultés majeures, car le temps de transfert devient inadmissible à cause du volume important des données. L'exploitation des images sur Internet, avec les services interactifs de distribution des images de télédétection, requiert de nouvelles approches de représentation.

La représentation de l'image doit satisfaire aux revendications de toute application concrète et doit être acceptable pour l'application dans les termes de la complexité du calcul de la tâche (c.-à-d. le nombre de pas ou les opérations arithmétiques demandées pour la résolution du problème).

Le transfert d'information entre les utilisateurs doit répondre à des exigences toujours plus aiguës en matière de rapidité, de facilité d'accès et de limitation des coûts. Selon Stephane Côté (CARTEL, 2009), les principales exigences des utilisateurs d'images sont :

- le format matriciel de représentation de données,
- la simplicité et l'avantage de l'utilisation d'images,
- la haute définition spatiale et la très haute définition spatiale,
- l'exactitude maximale,
- la disponibilité des données à différentes échelles,

- la disponibilité des données archivées.

En répondant à ces exigences, la question de traitement de données numériques est très actuelle. La nécessité de développer des nouvelles technologies de représentation, de transformation, d'archivage et de stockage des images de télédétection devient, de ce point de vue, immédiate.

1.2.1. Pourquoi compresser les données?

De nos jours, la puissance des processeurs augmente plus vite que les capacités de stockage et énormément plus rapidement que la bande passante des réseaux, car cela demande d'énormes changements dans les infrastructures de télécommunication. Pour pallier ce manque, il est courant de réduire la taille des données en exploitant la puissance des processeurs plutôt qu'en augmentant les capacités de stockage et de transmission des données.

Les étapes de compression et de décompression sont donc nécessaires pour éviter les limitations de la bande passante des réseaux (Figure 1.1).

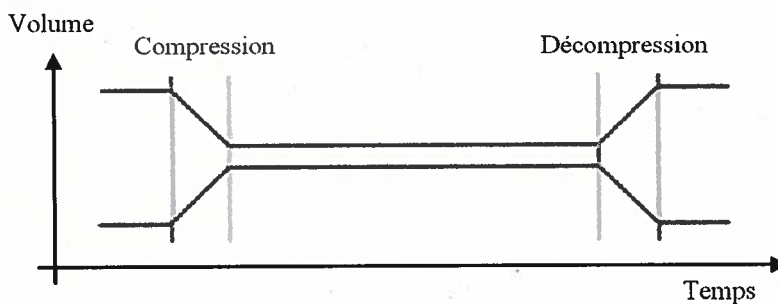


Figure 1.1 – Processus de la transmission des données par le canal de télécommunication (Gutter, 1995)

1.2.2. Qu'est-ce que la compression de données?

La compression consiste à réduire la taille physique des blocs d'information :

- un compresseur utilise un algorithme servant à optimiser les données en prenant en considération les caractéristiques propres au type de données à compresser;

- un décompresseur est donc nécessaire pour reconstruire les données originelles en inversant l'algorithme utilisé pour la compression.

Dans le cas de la compression symétrique, la même méthode est utilisée pour compresser et décompresser l'information, il faut donc la même quantité de travail pour chacune de ces opérations. C'est ce type de compression qui est généralement utilisé dans les transmissions de données (Guillois, 1996 ; Gutter, 1995).

1.2.3. Deux problématiques majeures de recherche

Les algorithmes actuels traitent l'information de façon séquentielle (c.-à-d. pas à pas). Cela a pour inconvénient principal d'augmenter le temps de calcul. En prévision des évolutions technologiques futures, où les systèmes informatiques parallèles fonctionneront comme le cerveau d'un être humain, il faut penser des algorithmes adaptés qui peuvent traiter l'information de façon parallèle et qui ont la possibilité de généraliser efficacement les connaissances du domaine étudié.

Nous avons posé deux problématiques majeures dans le domaine de la représentation des images de télédétection:

- a) La tendance des recherches dans le domaine de la représentation des images satellitaires réside dans la conception de méthodes hybrides qui combinent le traitement spatial et la transformation abstraite de l'image. Le problème de la compression d'images de THRS est qu'il a reçu moins d'attention de la part de la communauté scientifique (Gagnon et Lalonde, 1999);
- b) Dans le domaine du traitement des images de télédétection, le progrès le plus important est la découverte d'opérateurs de regroupement des pixels, permettant de séparer de façon optimale les informations de détail et de tendance. Le réseau de neurones propose une alternative intéressante avec les méthodes de regroupement de pixels par rapport aux méthodes classiques (Anonyme, 1991 ; Kalyani, 2009 ; Schalkoff 1992 ; Touzet, 1992 ; Zharkova and Jain, 2007).

1.3. Objectifs

Nous avons défini l'objectif général de la thèse comme étant la création d'un nouvel algorithme hybride à la base de la représentation adaptative des images de THRS par pyramide inverse. Ceci est rendu possible en utilisant le réseau de neurones artificiel pour trouver l'opérateur du regroupement des pixels, dans le but de réduire le volume d'image de THRS sans détérioration significative de la qualité visuelle de l'image restaurée.

Les sous-objectifs sont:

- a. Élaborer une nouvelle méthode de représentation adaptative des images satellitaires de THRS par pyramide inverse en utilisant des réseaux de neurones;
- b. Évaluer les résultats de l'expérimentation;
- c. Comparer les résultats obtenus aux méthodes existantes de compression d'images.

1.4. Hypothèses de la recherche

L'hypothèse générale du projet est que la création du nouvel algorithme adaptatif permettra de réduire davantage le poids des fichiers d'images de télédétection compressés en présentant une qualité visuelle semblable aux images restaurées par des algorithmes de compression existants.

Les sous-hypothèses sont :

- a. La représentation pyramidale d'images de THRS simplifiera un accès ciblé à l'information ;
- b. L'utilisation du réseau de neurones à rétropropagation permettra de découvrir l'opérateur efficace de regroupement des pixels pour séparer de façon optimale les informations de détails et de tendances.

1.5. Plan de la thèse

Le plan de la thèse est le suivant :

- Le deuxième chapitre est la revue de littérature dans le contexte du domaine de recherche de la thèse présentée.
- Le but du troisième chapitre est de présenter les méthodes utilisées au cours de ce projet. Nous discutons d'abord de deux méthodes efficaces du domaine de traitement d'images de télédétection : la transformation pyramidale et le traitement de données avec un réseau de neurones artificiels. Nous présentons ensuite la méthodologie développée en détaillant l'algorithme hybride proposé.
- Dans le chapitre quatre, les résultats de recherche publiés dans des revues à comité de lecture sont également présentés. Nous introduisons trois articles touchant les thématiques :
 - o du codage de la succession d'images de télédétection ;
 - o du codage d'images de télédétection de THRS ;
 - o de la transformation progressive d'images (PIT) de télédétection.

Nous prévoyons une certaine redondance de l'information pour ces trois articles en ce qui a trait à l'introduction et à la méthodologie. La raison principale réside dans le souci d'une présentation complète de la nouvelle méthode pour chaque article en soi.

- Le cinquième chapitre offre une discussion sur les résultats ; le sixième souligne la contribution scientifique et le septième discute de l'originalité du projet.
- Le huitième chapitre a trait à la conclusion du travail, présentée avec ses avantages et ses limitations.
- Le neuvième chapitre traite de quelques perspectives envisageables pour de futures recherches.

« La technologie de la compression des images est examinée comme la technologie de l'élargissement des possibilités. »

Guillois, J.-P.

2. Revue de littérature

2.1 Mise en contexte

La dernière décennie a vu l'expansion des techniques numériques dans de nombreux domaines de traitement de l'information, de la saisie, du stockage et du transport de cette information (Gonzalez and Woods, 2008 ; Burger and Burge, 2009 ; Gomasasca, 2009 ; Hoggar, 2006 ; Perry *et al.*, 2002 ; Provencher et Dubois, 2007 ; Read et Graham, 2002 ; Read and Graham, 2002). Pour économiser les ressources de stockage ou des canaux de communication, on cherche à compresser le signal numérisé. Cette double opération de numérisation et de compression du signal est aussi appelée codage de source.

La théorie de l'information (Shannon, 1948) formalise la notion de codage de l'information et définit des limites minimales théoriques, ou bornes asymptotiques, quant au nombre de bits nécessaires à la représentation d'une information.

Ces bornes sont données par l'entropie de la source dans le cas d'un codage sans perte ou par une fonction débit-distorsion dans le cas du codage avec perte. On cherche alors à approcher au mieux ces limites de performance ou encore à optimiser la qualité subjective du signal reconstruit pour un débit de transformation ou un volume d'information donné.

2.1.1. Contenu informatif des sources de données

Liée à la théorie de l'information (Shannon, 1948; Shannon et Weaver, 1963), l'entropie permet d'évaluer le contenu informatif d'une source. Cet indice a été utilisé par Chen *et al.* (1987) pour analyser le contenu informatif d'images Landsat dans le but de les compresser, ou encore par Tsagaris *et al.* (2005) dans le but de fusionner des bandes d'un capteur hyperspectral. Enfin, Bloch et Maitre (1994) proposent

l'emploi de l'entropie lors de l'étude de la redondance et de la complémentarité des sources dans un contexte de fusion de données.

Selon Shannon (1948), plus une source d'information est redondante, moins elle contient d'information. Ainsi, l'entropie sera maximale si les proportions des N classes d'une source sont toutes égales à $1/N$. Elle aura également tendance à croître avec l'augmentation du nombre de classes, ce qui représente une augmentation de l'incertitude. En d'autres termes, moins une observation est probable, plus son observation est porteuse d'information.

Conformément à Shannon la formule de l'entropie (H) d'une source S contenant N classes est :

$$H(S) = -\sum_{i=1}^N (p_i) * \ln (p_i) \quad (2.1)$$

où p_i : la proportion de la classe i dans la source.

2.1.2 Taux de compression

Le procédé inhérent à la compression des images consiste à réduire la dimension du fichier de données en ne retenant que les informations essentielles. Le taux de compression (CR - compression ratio, en anglais) est spécifié en utilisant un facteur de compression défini comme étant le rapport des dimensions du fichier non compressé et du fichier compressé. Les meilleurs procédés de compression sont ceux qui offrent le meilleur rapport entre un CR élevé et les informations essentielles retenues. Évidemment, cela diffère en fonction de l'application visée. Par exemple, dans une image binaire comportant seulement des informations textuelles, l'information pertinente n'implique que le texte, comparativement à une image médicale dont l'information pertinente à extraire peut consister en de détails très fins.

Dans le cadre de notre étude, nous nous sommes intéressés essentiellement à la compression avec perte (en ce qui a trait à l'image visuelle) afin d'obtenir des CR importants.

Le CR est défini par le rapport entre la taille de l'image à l'entrée et la taille de l'image à la sortie du codeur. Dans le cas général et, en particulier, dans le cas d'une image fixe à niveaux de gris, le CR d'un codeur est calculé par rapport à l'image codée. Le CR est alors :

$$CR = \frac{\text{taille_de_l'image_d'entrée}}{\text{taille_de_l'image_de_sortie}}; \quad (2.2)$$

où : la taille de l'image : hauteur * largeur * définition (8 bits) de l'image et la dimension de la taille de l'image est - un _ nombre _ d'éléments _ binaires .

2.1.3. Paramètre PSNR

Le rapport signal sur bruit SNR (Signal-to-Noise Ratio), en décibel (dB), est défini de la façon suivante (Antonin, 2000). C'est une mesure, en dB, de la fidélité de l'image compressée par rapport à l'image d'origine :

$$SNR = 10 \log_{10} \frac{\text{énergie_de_l'image_originale}}{\text{énergie_de_l'erreur_apportée_par_le_codeur}}; \quad (2.3)$$

Quand on utilise le rapport entre l'énergie maximale d'une image et l'énergie de l'erreur, on parle du rapport de signal sur bruit maximal PSNR (Peak Signal-to-Noise Ratio, en anglais). Il se base sur le calcul de l'erreur quadratique moyenne (MSE) entre l'image d'origine et l'image compressée. Dans le codage, la définition, c.-à-d. le nombre de bits pour coder un pixel, est typiquement 8 bits soit 256 (entre 1 et 256) niveaux de gris (ou entre 0 et 255 pour 8 bits) possibles par pixel (Antonin, 2000). Soit a l'image originale et \hat{a} l'image compressée, le PSNR est défini de la façon suivante :

$$PSNR = 10 \log_{10} \left(\frac{255^2}{\text{Variance}(a - \hat{a})} \right); \quad (2.4)$$

$$\text{Variance}(a - \hat{a}) = \frac{1}{\text{hauteur} * \text{largeur}} \sum_{ij} (a_{ij} - \hat{a}_{ij})^2; \quad (2.5)$$

où i et j – sont les coordonnées des pixels de l'image originale a et de l'image compressée \hat{a} ; la valeur de luminance maximale que peut prendre un pixel par exemple 255 (dans le cas d'une image 8 bits).

Un système de compression peut être défini en termes d'efficacité (CR) d'une part et en terme de qualité, d'autre part. Un CR plus important entraîne une qualité d'image moins bonne et inversement. Il convient de prendre, en suivant les applications recherchées, le codeur réalisant le meilleur compromis qualité – CR.

2.1.4. Similarité structurelle (SSIM)

SSIM (Structural SIMilarity) est une mesure de similarité entre deux images. Elle a été développée pour mesurer la qualité visuelle d'une image compressée, par rapport à l'image originale. L'idée de SSIM est de mesurer la similarité de *structure* entre les deux images, plutôt qu'une différence pixel à pixel comme le fait par exemple le PSNR. L'hypothèse sous-jacente est que l'œil humain est plus sensible aux changements dans la structure de l'image (Wang *et al.*, 2004 ; Wang *et al.*, 2009).

La métrique SSIM est calculée sur plusieurs fenêtres d'une image. La mesure entre deux fenêtres x et y de taille $N \times N$ est :

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2cov_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \quad (2.6)$$

avec

μ_x - la moyenne de x ;

μ_y - la moyenne de y ;

σ_x^2 - la variance de x ;

σ_y^2 - la variance de y ;

cov_{xy} - la covariance de x et y ;

$c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ - deux variables destinées à stabiliser la division quand le dénominateur est très faible ;

L - la dynamique des valeurs des pixels, soit 255 pour des images codées sur 8 bits;

$k_1 = 0,01$ et $k_2 = 0,03$ par défaut, qui contrôlent le dénominateur qui s'approche du zéro, (le choix de telles valeurs est défini pour éviter la situation instable de la formule 2.6).

2.2. Traitement d'images et vision artificielle

Le traitement d'images est souvent une étape préliminaire à l'extraction des caractéristiques (voir annexe 2) (Gagnon et Lalonde, 1999). Il englobe les domaines de la restauration, de l'accentuation et de la compression des « données images », tels qu'on les conçoit dans les applications nécessitant du débruitage (ex. la réduction du bruit de chatoiement dans les images radars et acoustiques), de la compensation de la distorsion causée par un système optique inadéquat ou de la stabilisation de séquences vidéo.

Le traitement d'images ne fait pas appel, *a priori*, à des techniques d'intelligence artificielle comme la vision par ordinateur (voir annexe 3). Le système de vision des humains (voir annexe 4) fait un travail remarquable à cette tâche, sans même y penser (Hubel, 1988). L'implémentation d'une capacité semblable dans des algorithmes informatiques pour traiter des images de THRS est une tâche stimulante.

Dans le cas des images de télédétection, nous voudrions extraire les renseignements spatiaux d'une image à différents niveaux : des détails fins dans les zones locales aux caractéristiques globales de l'image.

2.3. Deux approches potentielles d'utilisation de l'image de télédétection

L'utilisation potentielle de l'image de télédétection relève de deux approches : l'interprétation visuelle et l'analyse à des fins quantitatives. La première est de nature subjective dans la mesure où elle repose sur notre système psychosensoriel. La deuxième concerne l'exploitation de l'image à des fins de mesure et d'analyse de paramètres physiques, chimiques ou biologiques (Tonye, 2000).

Un compromis reste à trouver entre les impératifs de fidélité, de qualité de l'information et de rapidité de transfert. Les diverses méthodes hybrides actuelles de décomposition et de compression de l'information apportent une réponse à ces besoins (Barlaud et Labit, 2002 ; Salomon, 2004 ; Alföldi, 1978 ; Aizzi *et al.*, 1996 ; Dasarathy, 1995 ; Gibson et Power, 2000 ; Kou, 1995).

Une des caractéristiques principales de l'accès et du transfert d'information étant la rapidité, il y a lieu de repenser ces mécanismes avant tout processus de compression. Les caractéristiques du système doivent permettre un accès ciblé à l'information, de sorte que seule l'information pertinente pour une échelle et pour une région d'intérêt donnée soit transférée vers l'utilisateur. Il est ainsi possible de limiter les volumes d'information à gérer et, de ce fait, de réduire le temps et les coûts consécutifs (Pointet *et al.*, 2003).

2.4. Rappel des techniques de représentation

Pour résoudre le problème de l'expédition des fichiers ainsi que l'archivage des images de THRS pour de longues périodes, on peut avantageusement faire appel aux techniques de représentation des images, car elles permettent de réduire la taille des données tout en éliminant la redondance des informations qu'elles contiennent (Kunt, 1996 ; Isar *et al.*, 2002).

Souvent, l'espace de l'image n'est pas toujours le plus approprié pour effectuer le traitement ou l'analyse d'une image. Il est souvent plus indiqué de transformer une image dans un espace abstrait pour lequel les données transformées sont plus adaptées au problème (Chen et Zhang, 1999). Les développements mathématiques récents (par exemple, l'utilisation des réseaux de neurones artificiels) et les méthodes numériques jouent alors un rôle déterminant (Hérault, 1994 ; Prasard et Prasanna, 2008 ; Hu et Hwang, 2002).

2.5. Compression d'image

La compression d'image s'adresse à la problématique de réduction de la quantité de données exigées pour représenter une image. Le procédé inhérent à la compression d'image consiste à réduire la dimension du fichier de données en ne retenant que les informations essentielles. La compression est accomplie par le retrait d'un ou

plusieurs des trois redondances fondamentales (Gonzalez et Wood, 2008 ; Burger et Burge, 2009):

- la **redondance du code**, qui est présente quand la longueur du mot de code n'est pas optimale;
- la **redondance d'interpixel**, relations entre les pixels d'une image;
- la **redondance psychovisuelle**, la partie des données absentes imperceptible par le système visuel de l'être humain.

Il existe deux types de méthodes de compression : avec ou sans perte d'information. Avec les méthodes sans perte, aucune donnée n'est perdue lors de la phase de compression. Ces méthodes sont nécessaires pour certaines applications telles que les images médicales, les documents juridiques, etc. Par exemple, si le médecin fait un diagnostic erroné, l'erreur peut être fatale pour le patient.

2.5.1. Compression sans perte (les méthodes réversibles)

Les méthodes réversibles sont basées sur le dénombrement statistique des données et sur le calcul de leur entropie.

Le principe général de la compression sans perte consiste à assigner le plus petit code binaire au symbole le plus fréquent, de manière à réduire le volume d'information. Ce principe est connu sous le nom d'encodage à longueur variable (*Variable-Length Coding* VLC) ou encodage d'entropie (Karam, 2000). Il a donné naissance à plusieurs techniques d'encodage, notamment l'encodage de Huffman, le code RLE (*Run length Encoding*) ou le code LZW (*Lempel-Ziv-Welch*) (Kovacs et Ranganathan, 1995 ; Marsault, 1995 ; Held et Marshall, 1996) et l'encodage arithmétique. Avec de tels algorithmes, la reconstitution de l'image est fidèle, mais le CR est faible. En effet, ce dernier est de l'ordre de 10 %.

Un cas particulier de données d'images se prêtant particulièrement bien à la compression sans perte est celui des données thématiques, telles que les cartes numérisées ou les images classées de télédétection. Elles ont l'avantage d'être

constituées d'un nombre restreint de valeurs de pixel; il est donc aisé de les compresser sans perte, même à des taux élevés.

2.5.2. Compression avec perte (les méthodes irréversibles)

Les méthodes irréversibles (compression avec perte) prennent en considération les propriétés de la perception visuelle (Jolion, 2001). Compte tenu des limites de la perception de l'œil, une quantification appropriée de l'image peut conduire à une compression élevée, puis à une reconstruction acceptable de l'image. Cependant, il apparaît dans ce cas des artefacts se traduisant par des effets de bloc, de bordure et d'image floue, lors de la reconstruction.

Les méthodes de compression avec perte impliquent la perte d'une certaine quantité de données (Ameur *et al.*, 2002). L'image finale ne comporte pas toutes les informations initiales qui étaient présentes dans l'image originale. Le principe de base consiste à éliminer l'information qui n'affecte pas ou peu l'aspect visuel (Chen et Wang, 2005).

Parmi les techniques de compression de données avec perte, notons le codage à longueur variable (RLE) (runlength coding algorithm applied to gray-level images (*gray-level run-length*)), le codage par la troncation de bloc (*block truncation BTC*), la quantification vectorielle (vector quantization VQ), la prédiction différentielle (differential predictive) et le codage transformé (transform coding) (Campbell, 2002 ; Jensen, 2005 ; Rao et Yip, 2001).

Nous pouvons citer les méthodes qui font appel à la transformée de Fourier FFT (*Fast Fourier Transform*), à la transformée de cosinus discrète DCT (*Discret Cosinus Transform*), à la JPEG (*Joint Photographic Experts Group*) et à la transformée en ondelettes DWT (*Discret Wavelette Transform*) (Kunt, 1984 ; Kim, 1991 ; Cohen, 1992 ; Kovas et Ranganathan, 1995 ; Mallat, 1989 ; Moulin, 2000).

Avec ces différentes techniques, il est possible d'obtenir un CR de 20 pour des images complexes et pouvant atteindre 200 pour des images simples. La qualité visuelle reste acceptable pour des taux encore supérieurs.

Le standard JPEG est une forme du codage transformé qui utilise la transformée en cosinus discrète. Les techniques de compression des images par transformée en ondelettes (JPEG2000) effectuent la compression d'une image en un tout (en segmentant l'image en composantes fréquentielles). La compression des données par transformée en ondelettes permet d'obtenir un CR pouvant atteindre 350 pour des images simples.

Les résultats obtenus par Pointet *et al.*, 2003, montrent les limites rapidement atteintes des algorithmes standards tels que JPEG, JPEG2000, quand la taille des images augmente, particulièrement pour les images de télédétection de THRS.

2.6. Algorithmes de compression actuels

On peut diviser les algorithmes de compression actuels aux formats courants et spécialisés. Il y a deux formats courants, respectivement ancienne et nouvelle génération : la norme JPEG Standard et JPEG 2000. Deux formats spécialisés utilisant une approche de compression par transformée en ondelettes sont actuellement exploités dans le domaine de la télédétection et de la photogrammétrie : MrSid (*MultiResolution Seamless Image Database*) de la firme LizardTech et ECW (*Enhanced Compression Wavelets*) de la compagnie ERMapper (Pointet *et al.*, 2003).

Malgré leur efficacité, avec des CR acceptables, ces algorithmes nécessitent de grandes dépenses calculatoires (Perry *et al.*, 2002). Notre objectif de recherche est la création d'un algorithme qui prend en compte le système de perception visuelle humaine (*Human Vision System HVS*) (Jolion, 2001 ; Hubel, 1988) tout en diminuant le temps de calcul.

Nous avons trouvé dans la littérature de nombreuses méthodes de compression des données (Isar *et al.*, 2002 ; Guillois, 1996 ; Hoggar, 2006 ; Gutter, 1995; Kou, 1995), mais pour la compression d'images de télédétection, le nombre d'œuvres récentes est faible (Caloz et Collet, 2001; Pointet *et al.*, 2003). Souvent, les auteurs tentent d'appliquer les méthodes classiques de compression des données aux images de télédétection.

Plusieurs auteurs traitent de la compression et de la transmission d'images multibandes (Ameur *et al.*, 2002 ; Guillois, 1996 ; Pointet *et al.*, 2003).

Les méthodes classiques pourraient être classées selon les groupes principaux suivants (Isar *et al.*, 2002; Kou, 1995; Rao et Rip, 2001; Salomon, 2004) :

- déterministe et transformée statistique linéaire orthogonale (transformée en cosinus discrète (DCT), transformée de Fourier discrète (DFT), transformée Walsh-Hadamard (WHT) et transformée de Karhunen-Loève (KLT), analyse en composantes principales (ACP); décomposition en valeurs singulières (SVD) (Kou, 1995) ;
- transformée en ondelettes discrète (DWT), embedded zero-tree ondelettes (EZW) (Topiwala, 1998) ;
- transformée basée sur la prédiction (prédiction linéaire (LP)) ;
- quantification vectorielle (VQ) ;
- transformée fractale (FT) ;
- décompositions, basées sur les pyramides diverses : la pyramide Gaussien (GP) et la pyramide de Laplace (PL) (Burt et Adelson, 1983) .

L'analyse de ces méthodes montre que celles qui sont habituellement utilisées, pour la compression d'images, sont la transformée déterministe orthogonale et la prédiction linéaire avec des coefficients fixes.

Par exemple, le standard JPEG est basé sur la transformée en cosinus discrète (DCT) et en prédiction linéaire (LP), alors que le standard JPEG2000 est basé sur la transformée en ondelettes discrète (DWT).

Les données sont compressées avec le codage entropique (CE), mettant en œuvre une combinaison de différentes méthodes de codage sans perte (codage par plage (RLE), codage arithmétique (AC) et codage Lempel-Ziv-Welch (LZW).

Les méthodes de transformation comme DCT, DWT et LP sont déterministes et les valeurs des coefficients des matrices ne dépendent pas du contenu de l'image traitée. Pour cette raison, l'efficacité de la compression de telles méthodes est faible lorsque la corrélation entre le contenu de l'image et les fonctions de transformation correspondantes est faible.

De plus, pour des méthodes telles que DCT (JPEG) et DWT (JPEG2000), il faut avoir de nombreuses fonctions de transformation pour la compression et pour la décompression des données des deux côtés (codeur et décodeur).

Par exemple, la transformée en ondelettes présente un grand intérêt si l'opération inverse, la reconstruction du signal par une somme pondérée d'ondelettes, existe. Un jeu d'ondelettes quelconque est incapable de remplir cette fonction. Pour que la reconstruction soit possible, il est nécessaire que la décomposition soit réalisée sur une famille d'ondelettes indépendantes entre elles, constituant ainsi une base orthonormée ou orthogonale.

Les transformées statistiques sont plus efficaces que les déterministes, mais elles ont une complexité de calcul plus élevée.

Les méthodes fractales ont un inconvénient similaire. Par ailleurs, elles ne sont pas assez efficaces pour les images contenant des régions obscures de la texture sont traitées.

Les décompositions pyramidales de l'image sont généralement réalisées avec des filtres numériques de décimation fixe et avec l'interpolation des coefficients ; elles utilisent aussi un des types de la transformée, par exemple la DCT multiple. Autrement dit, ces méthodes ne sont pas bien adaptées au contenu de l'image.

2.7. Architecture des algorithmes de compression

Les algorithmes de compression d'images présentent plus ou moins la même architecture : une première étape de décorrélation de l'information suivie d'opérations de quantification et de codage (Figure 2.1). Le codage entropique permet de compresser sans perte d'information (Shannon, 1948). En revanche, l'opération de quantification génère des pertes d'information. Il est important, dans ce cas, de limiter au maximum les dégradations selon des critères psychovisuels. Des critères objectifs sont des mesures de performance débit-distorsion, permettant d'estimer les dégradations (Guillois, 1996).

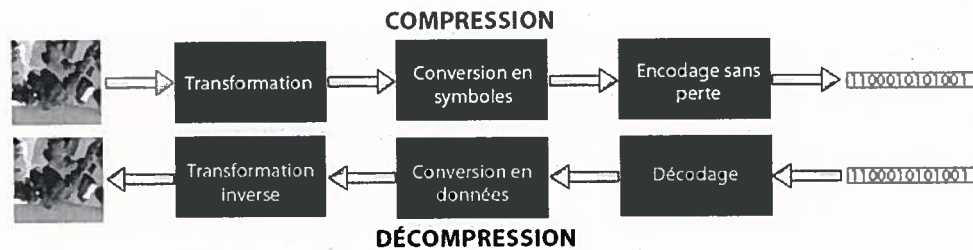


Figure 2.1 – Processus global de la compression – décompression d'images (Pointet *et al.*, 2003).

2.8. Généralités sur les méthodes de représentation d'images

2.8.1. Transformation pyramidale (ou décomposition multiéchelle)

Pourquoi la pyramide? Il existe de nombreux avantages à utiliser la représentation pyramidale d'une image. La motivation principale est de pouvoir manipuler, traiter, analyser l'information à des échelles spécifiques, sans interférer ou ajouter un temps de calcul inutile en traitant des informations non pertinentes au problème (Burt et Adelson, 1983 ; Adelson et Simoncelli, 1981 ; Kountchev *et al.*, 2004 ; Hwang et Derin, 1995 ; Tzovaras et Strintzis, 2002 ; Kountchev et Rubin, 2000).

Finalement, ce genre de représentation emprunte à la perception humaine des images. Les niveaux hauts (résolutions grossières) contiennent les caractéristiques globales de l'image, un peu comme le ferait l'humain qui regarde initialement une scène dans son ensemble en faisant abstraction des détails, pour ensuite converger vers les points d'intérêt (résolution fine) (Figure 2.2).

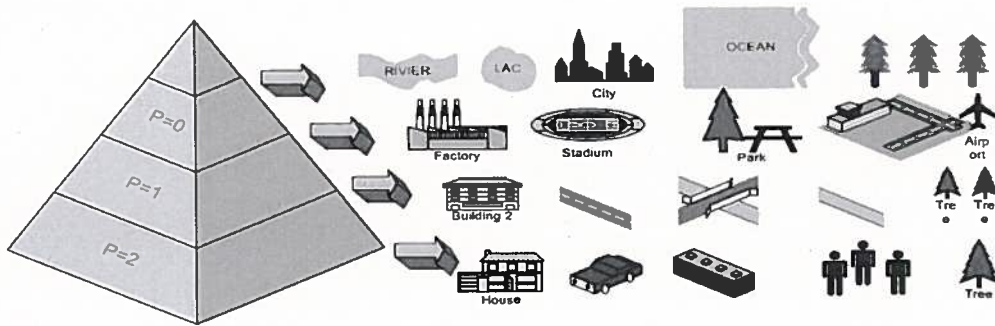


Figure 2.2 – Correspondance des niveaux de la pyramide vers des objets que l'on peut définir

En traitement d'images de télédétection, on crée une pyramide en combinant, par une opération donnée, des groupes de pixels voisins dans une image pour produire un nouveau pixel de résolution plus grossière et donc, une image de résolution et de dimensions réduites. Puisque les pixels sont combinés en groupes voisins, l'information spatiale est propagée à travers la pyramide, mais à des résolutions décroissantes. Ce processus est répété sur cette nouvelle image et successivement sur les autres. L'image initiale peut être réduite à un seul pixel qui donne une mesure globale de la brillance de l'image (Figure 2.3).

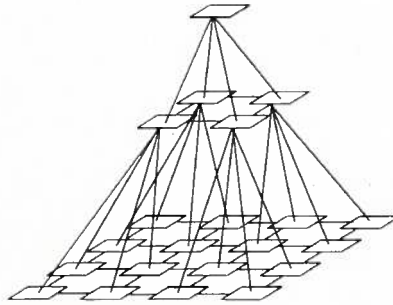


Figure 2.3 – Représentation des données numériques par la méthode classique de la pyramide
(Gonzalez et Woods, 2008)

Cette structure sépare donc l'information dans une image en différents « canaux d'échelles », les structures grossières étant représentées sur une grille fine et les détails fins sur une grille grossière.

La présentation pyramidale des données permet d'appréhender l'image de façon « non causale » par rapport aux parcours linéaires classiques. L'idée générique réside dans un raffinement de plus en plus précis de l'information codée (Barlaud et Labit, 2002).

2.8.2. Traitement de données par un réseau de neurones artificiels

Pourquoi le réseau de neurones (RN)? Le RN est un outil largement répandu depuis les vingt dernières années.

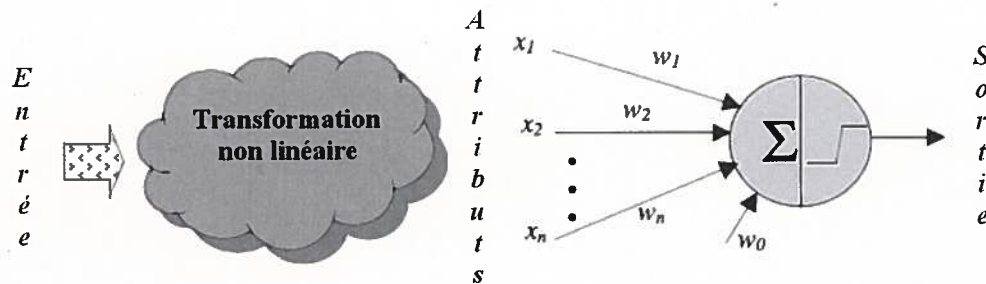


Figure 2.4 – Modèle du perceptron. (Hu et Hwang, 2002)

Les réseaux de neurones (RNs) sont considérés comme appartenant à la classe des algorithmes connus sous le nom des techniques de calculs intelligents des systèmes du traitement des images (Perry *et al.*, 2002). Un réseau de neurones est un ensemble d'unités de calcul (perceptron : Figure 2.4) interconnectées, ayant la propriété de simuler l'activité neuronale du cerveau humain (apprentissage, généralisation-prédiction, mémoire distribuée, classification, régression, etc.).

De nombreux RNs ont été proposés dans la littérature et peuvent être caractérisés sous différents angles (Saad et Wunsch, 2007 ; Luo, 1997 ; Gelenbe *et al.*, 1996) :

- leur architecture (l'arrangement spécifique des unités de calcul), la plus populaire étant le réseau de neurones multicouches;
- la règle d'apprentissage, qui dépend fortement de l'architecture;
- le mode d'apprentissage : soit dirigé, soit non dirigé.

L'apprentissage ne consiste pas à l'accumulation de toutes les situations possibles (l'augmentation de la base de données à l'infini), mais correspond à la généralisation des informations (voir Glossaire), en trouvant les meilleurs paramètres du RN pour une image du domaine étudié. Ainsi, si on traite le même type d'images, c.-à-d. images de télédétection, on peut utiliser le même RN qui est déjà entraîné, ce qui permet d'éliminer le temps d'apprentissage pour les autres images.

En outre les avantages du RN sont :

- leur large structure parallèle ;
- leur haut degré de corrélation ;
- la capacité d'apprentissage et l'auto-organisation.

2.8.3. Discussion sur les méthodes utilisées

Les images de télédétection étant de grande taille, leur représentation par la pyramide permet tout d'abord de visualiser rapidement le contenu général d'une scène. Elle permet, ensuite, à un utilisateur de sélectionner précisément la zone d'étude (une partie de l'image) qu'il désire recevoir avec la meilleure qualité (Aizzi *et al.*, 1996; Hwang et Derin, 1995).

Dans ce projet, nous utilisons les RNs afin d'optimiser le temps de calcul de la représentation pyramidale multiéchelle. La technique du traitement avec le RN permettra d'initialiser la reproduction de l'image en utilisant l'approche approximative. À l'aide du mode d'apprentissage du réseau de neurones, nous obtiendrons des coefficients de la matrice de transformation plus précis et associés au domaine d'étude. Cela permettra de visionner l'image compressée avec une qualité admissible et, ainsi, d'utiliser cette amélioration pour chaque niveau de la pyramide.

Le chapitre 3 présente la méthodologie de la recherche proposée.

3. Méthodologie

Pour appliquer notre méthode hybride de compression d'images de télédétection, il faut utiliser la combinaison de deux méthodes :

- la méthode de représentation des données par pyramide inverse (Aizzi *et al.*, 1996; Tzovaras et Strintzis, 2002; Kountchev *et al.*, 2004; Gagnon et Lalonde, 1999);
- la méthode de traitement des données avec un réseau de neurones (Hérault, 1994; Groupe Réseaux de neurones, 1991; Gagnon et Lalonde, 1999).

La première approche présente une décomposition multiéchelle inverse de l'image qui permet d'effectuer un accès ciblé à l'information, de sorte que seule, l'information pertinente pour une échelle et pour une région d'intérêt données soit transférée vers l'utilisateur.

La deuxième approche réalise le codage adaptatif d'images en cherchant les coefficients de la matrice de transformation en fonction du contenu de l'image originale.

Les étapes de la méthodologie développée sont présentées dans l'organigramme de la figure 3.1.

3.1. Représentation pyramidale inverse différentielle

La compression de cette représentation vient du fait que l'entropie des images d'erreur (des images différentielles) est plus faible que celle des images d'origine. De plus, comme l'histogramme de ces images d'erreur est concentré autour de 0, il est facile d'utiliser un codeur entropique avec des mots de largeur variable.

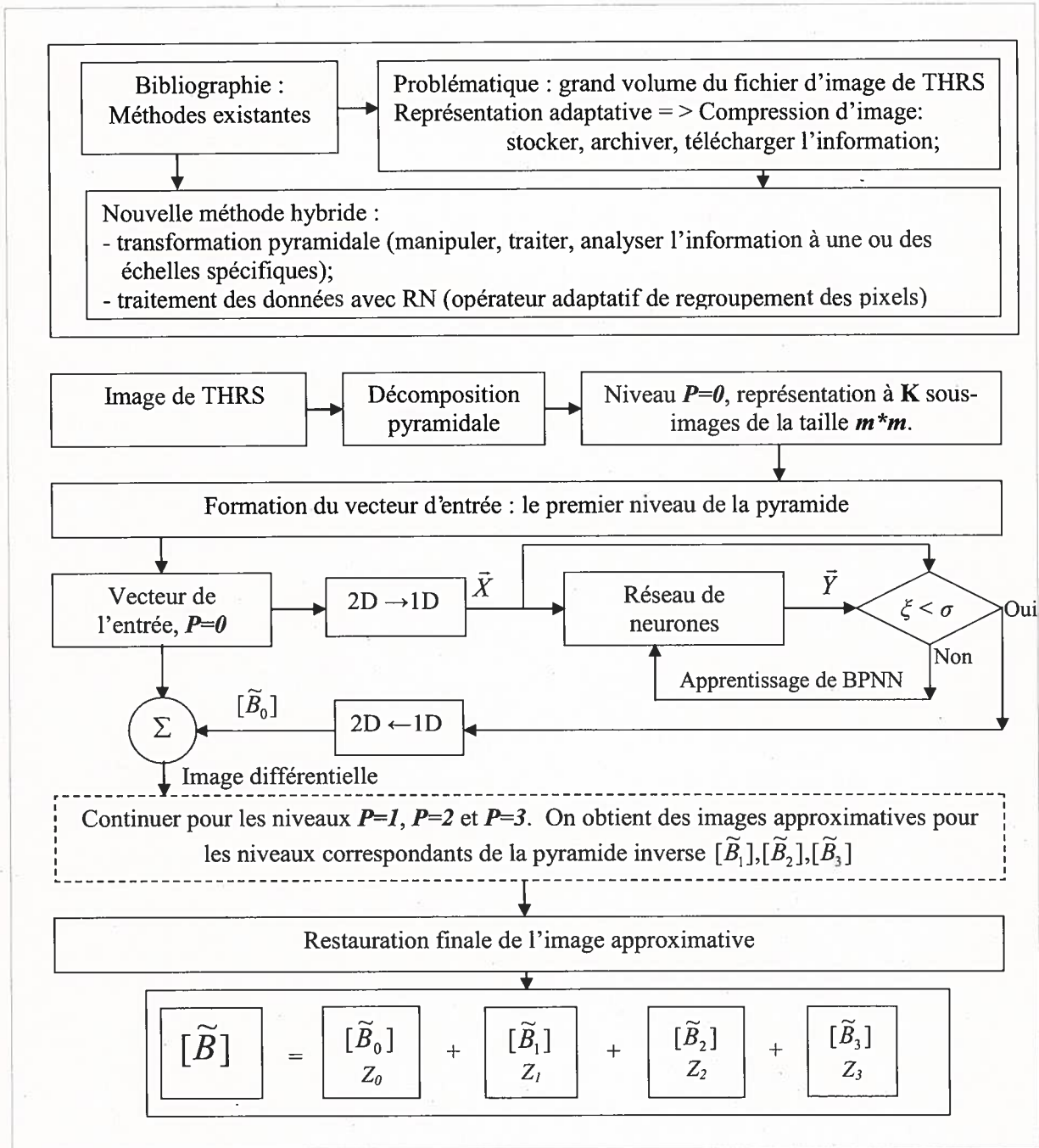


Figure 3.1 – Organigramme de la démarche scientifique

3.1.1. Décomposition d'images par la Pyramide inverse différentielle

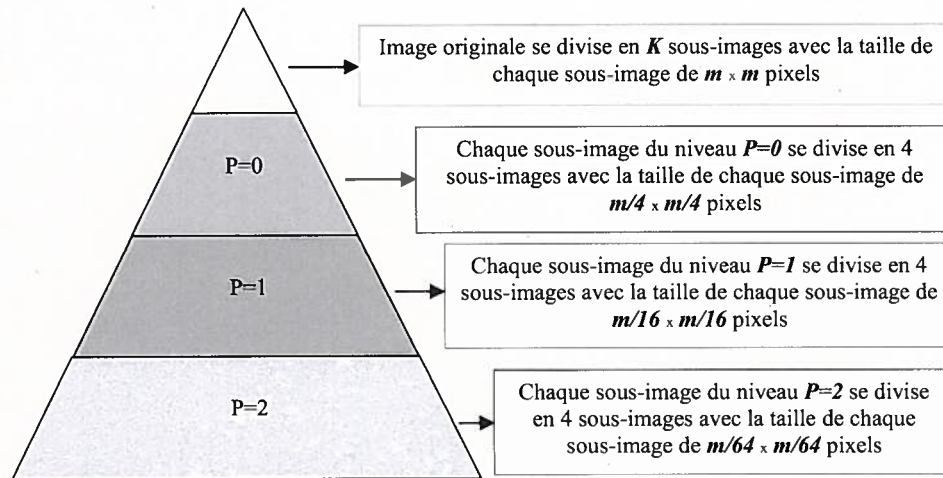


Figure 3.2 – Décomposition pyramidale d'images : le niveau plus bas de la pyramide ($P=2$) et le niveau plus haut de la pyramide ($P=0$).

La nouvelle décomposition, nommée la Pyramide inverse différentielle (PID) (Kountchev *et al.*, 2000 ; Kountchev *et al.*, 2004), est du type non orthogonal (Figure 3.2). Avant d'effectuer la décomposition, l'image numérique doit être divisée en sous-images de taille $N \times N$ ($N=2^n$) pixels. Chacune de ces sous-images se représente par une pyramide inverse avec p_0 couches. La valeur de p_0 est dans l'intervalle $1 < p_0 \leq n$ (ici le cas $p_0=n$ correspond donc à la décomposition pleine sans erreur).

Pour la détermination du sommet de cette pyramide, on utilise l'information qui contient la sous-image entière. La couche inférieure de la pyramide s'appuie sur l'information de quatre sous-images issues de la division par quatre d'une image différentielle correspondant à la différence entre la sous-image et son approximation. En continuant vers la base de la pyramide, la couche suivante est déterminée en divisant chacune des sous-images issues de la division précédente, en quatre sous-images (Figure 3.3).

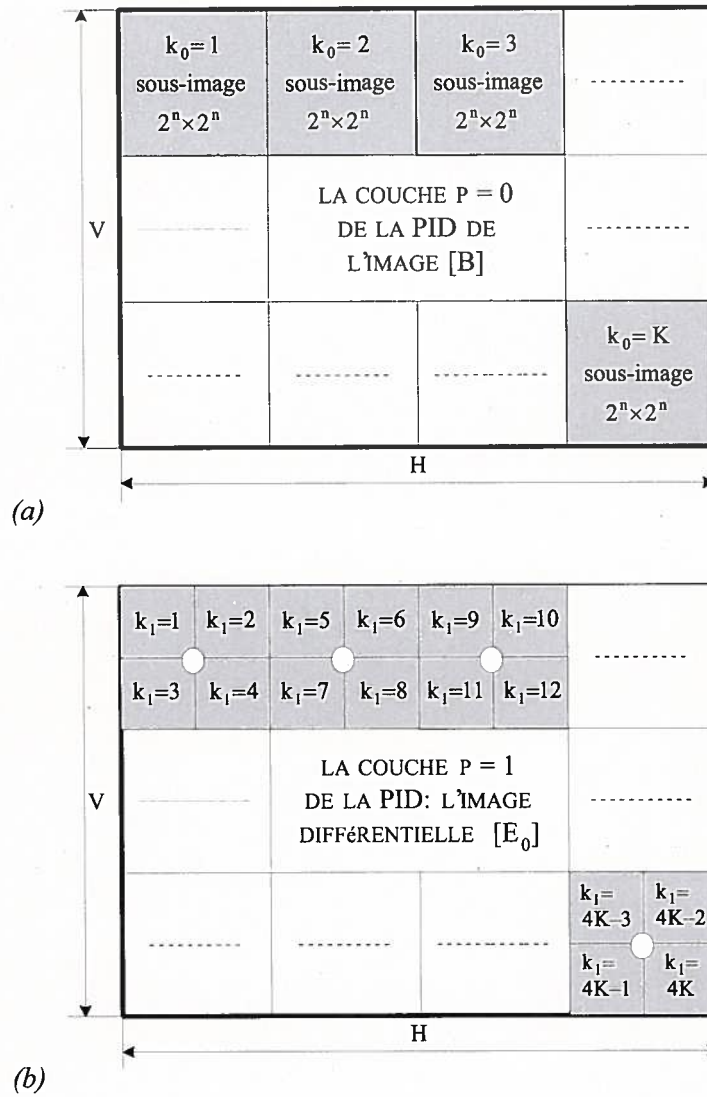


Figure 3.3 – Couches $p = 0$, $p = 1$ de la PID pour l'image $[B_{i,j}]$ de taille $H \times V$ pixels :
a. Division de l'image en K sous-images de taille $2^n \times 2^n$ pixels dans la couche $p=0$;
b. Division de chaque sous-image de la couche $p=0$ (dans les cercles) à quatre sous-images de taille $2^{n-1} \times 2^{n-1}$ pixels dans la couche $p=1$ (Kountchev et Rubin, 2000)

La matrice de l'image $[B]$ est divisée en K sous-images $[B_{k_0}]$, repérées par leur numéro k_0 . Alors, la matrice d'une sous-image $[B_{k_0}]$ peut être représentée par une décomposition matricielle avec $(n+1)$ couches, appelée la PID :

$$[B_{k_0}] = [\tilde{B}_{k_0}] + \sum_{p=1}^{n-1} [\tilde{E}_{k_{p-1}}] + [E_{k_n}] \quad (3.1)$$

pour $k_p = 1, 2, \dots, 4^p K$, quand $p = 0, 1, \dots, n-1$.

Ici k_p est le numéro de la sous-image correspondante dans la couche $p = 0, 1, \dots, n-1$, contenant $4^p K$ sous-images de taille $N_p \times N_p$ pixels ($N_p = 2^{n-p}$). Les composantes $[\tilde{B}_{k_0}]$ et $[\tilde{E}_{k_{p-1}}]$ ($p > 0$) représentent les approximations des matrices $[B_{k_0}]$ et $[E_{k_{p-1}}]$, définies ci-après. La composante $[E_{k_n}]$ représente l'erreur d'approximation, ou le résidu de la décomposition, contenant $(n-1)$ composantes. La matrice $[E_{k_{p-1}}]$ décrit une sous-image différentielle avec le numéro k_{p-1} dans la couche $p = 1, 2, \dots, n-1$ de la pyramide, déterminée par la relation :

$$[E_{k_{p-2}}] = [E_{k_{p-1}}] - [\tilde{E}_{k_{p-2}}] \quad (3.2)$$

avec condition initiale $p = 1$:

$$[E_{k_0}] = [B_{k_0}] - [\tilde{B}_{k_0}]. \quad (3.3)$$

La matrice $[E_{k_{p-1}}]$ de la sous-image différentielle dans la couche p se divise en $4^p K$ sous-images $[E_{k_p}]$ et pour chacune d'elles, on calcule une matrice correspondante d'approximation $[\tilde{E}_{k_p}]$. L'ensemble de toutes les sous-images $[\tilde{E}_{k_p}]$ pour $k_p = 1, 2, \dots, 4^p K$ détermine la composante suivante $(p + 1)$ de la décomposition (3.1). Alors, pour calculer la composante $(p + 2)$, on doit préalablement déterminer la matrice

différentielle, puis on répète toutes les opérations d'une façon analogue à celle-ci pour la composante précédente.

Les composantes $[\tilde{B}_{k_0}]$ et $[\tilde{E}_{k_p}]$ peuvent être représentées par deux types de modèle d'approximation : polynomial et spectral. Chacun d'eux se détermine par un nombre minimal de coefficients, calculés sur la base de l'erreur quadratique moyenne (MSE) de l'image d'entrée et de l'image restaurée pour de couche correspondante de la pyramide.

On applique d'abord, sur les coefficients du polynôme approximatif ou sur les coefficients spectraux retenus pour chacune des sous-images dans la couche p de la PID, une quantification et, ensuite, un codage entropique.

3.1.2. Codage de coefficients de la PID

On applique alors, sur les coefficients de la matrice de transformation, un codage adaptatif à longueur variable des plages et le codage de Huffman à toutes les couches de la PID. Le CR et la qualité visuelle de l'image restituée dépendent donc, notamment, du nombre de couches p_0 de la PID, du type et du nombre des coefficients choisis pour une sous-image, de la matrice de quantification et du type de transformation pour chaque couche p .

En pratique, il existe de nombreux cas où il est nécessaire de réaliser une transmission progressive d'images par l'augmentation de la résolution d'image combinée à la base d'image précédente. La structure de la PID est très bien adaptée à cette exigence, puisqu'elle permet une transmission progressive des images, couche par couche. Les données peuvent être mises en paquets en conformité avec la méthode choisie de restitution de l'image – en conservant l'augmentation progressive de la résolution ou de la qualité de l'image restaurée.

3.1.3. Décodage de données compressées par la PID

Après la transmission de l'information pour toutes les couches p de la PID sur les données compressées suivantes (3.1), on exécute un décodage d'Huffman et de longueur variable des plages, une quantification inverse des coefficients, puis une restitution d'image selon :

$$B'(i, j) = \tilde{B}_{k_0} + \sum_{p=1}^{p_0-1} \tilde{E}_{k_{p-1}}(i, j) \quad (3.4)$$

pour $i, j = 0, 1, 2, \dots, 2^n - 1$

Ici $B'(i, j)$ est le niveau de gris du pixel (i, j) de l'image restituée.

La décomposition d'images par la PID est présentée sur la figure 3.4

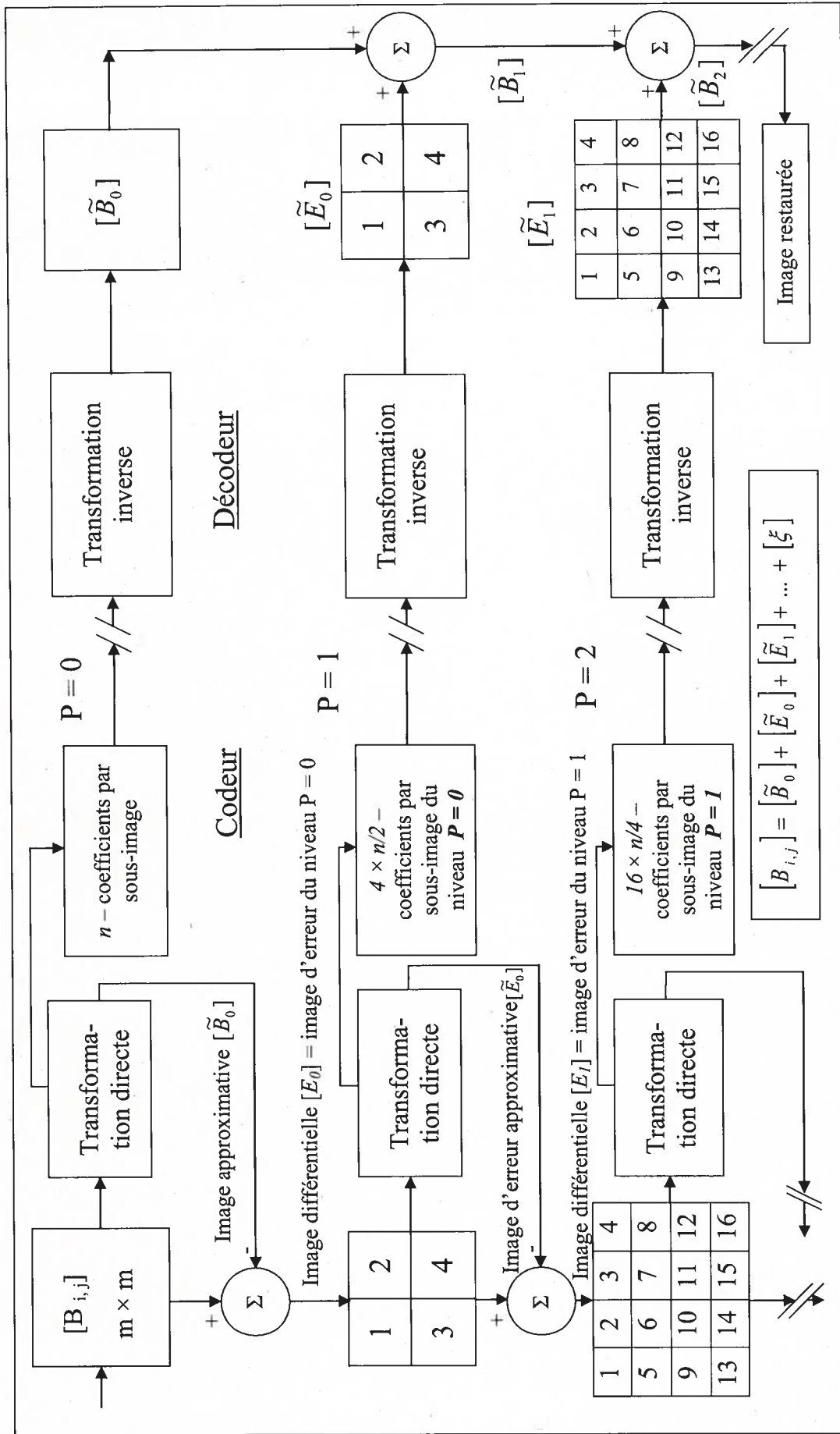


Figure 3.4 – Décomposition d'images par la pyramide inverse différentielle

3.2. Réseau de neurones multicouche

Nous avons étudié différents types des réseaux de neurones (Hérault, 1994; Hu et Hwang 2002; anonyme (1991); Touzet, 1992; Schalkoff, 1992) (voir annexe 5) et pour notre projet, nous avons décidé d'utiliser un réseau de neurones à rétropropagation (BPNN- *Back Propagation Neural Network*) avec l'architecture multicouche (MLP - *Multi Layers Perceptron*) qui permet de répondre aux besoins de notre méthode. La principale limite de ce type de RN est le temps de calcul, mais il peut largement contribuer à la qualité et à l'interprétation correcte des images.

Rappelons les qualités du BPNN :

- la restitution adéquate dans le domaine de l'imagerie;
- la baisse des efforts calculatoires (des opérations récurrentes);
- l'adaptation au contenu de l'image et du domaine de l'étude (à l'aide de l'étape d'apprentissage).

Nous avons choisi un réseau de neurones à trois couches (Figure 3.5) : une couche d'entrée, une couche cachée et une couche de sortie. Le réseau de neurones compte le même nombre de neurones pour la couche d'entrée et pour la couche de sortie, situées au même niveau de la pyramide. La raison d'un tel choix est justifiée par la nécessité de comparer l'image restaurée à l'image originale avec une résolution identique pour les deux images. Le nombre de neurones de la couche cachée est défini en considérant la qualité de l'image restaurée et le niveau de compression à un certain niveau de la pyramide.

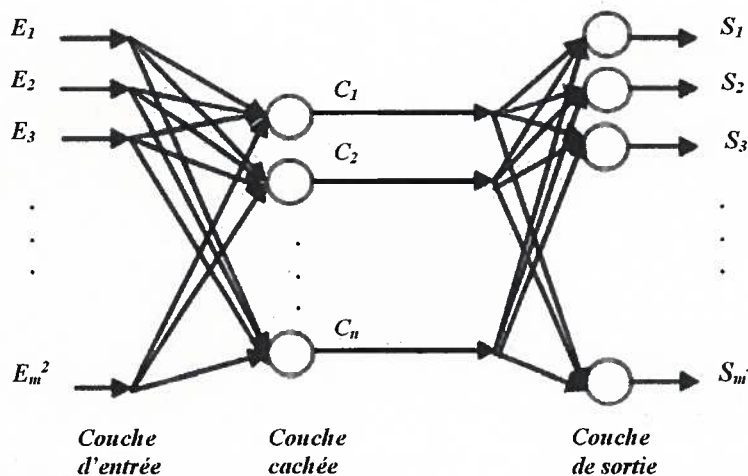


Figure 3.5 – Réseau de neurones multicouche

Remarquons que pour différents niveaux de la pyramide, nous avons utilisé des réseaux de neurones avec des quantités de neurones différentes.

La relation entre le vecteur d'entrée du réseau de neurones et le vecteur de sortie est présentée par l'équation suivante :

$$\vec{E} \approx \vec{C} \left[W_{n,m^2} \right] \approx \vec{S} \quad (3.5)$$

où $\vec{C} \ll \vec{E}$, et \vec{C} - le vecteur des valeurs des neurones de la couche cachée ; \vec{E} - vecteur d'entrée, $[W_{n,m^2}]$ - matrice des poids synaptiques et des seuils des neurones de la couche cachée et de la couche de sortie, \vec{S} - vecteur restauré (Figure 3.5).

Un avantage important de cette méthode réside dans le fait que nous utilisons la capacité d'apprentissage du réseau de neurones. Le réseau de neurones est capable de généraliser l'information pendant l'étape d'apprentissage. Après cette étape, le réseau de neurones est prêt au travail dans le domaine d'étude ciblé, même avec des données qui n'ont pas servi durant l'étape d'apprentissage.

Le tableau 2.1 représente l'approche pyramidale de l'image d'entrée 512 x 512 pixels et la correspondance de l'architecture du réseau multicouche de neurones en fonction de nombre de couches de la pyramide.

Tableau 2.1 – Structure pyramidale de 10 niveaux du réseau de neurones à trois couches pour l'image de résolution 512 x 512 pixels

	Niveau de la pyramide	Nombre de sous-images	Taille de sous-image	Structure de sous-image	Nombre de neurones pour les couches : (entrée : cachée : sortie)	Structure de données dans la couche cachée
P0	1	1	512 x 512	1 x 1	262144 x 512 x 262144	1 x 512
P1	2	4	256 x 256	2 x 2	65536 x 256 x 65536	4 x 256
P2	3	16	128 x 128	4 x 4	16384 x 128 x 16384	16 x 128
P3	4	64	64 x 64	8 x 8	4096 x 64 x 4096	64 x 64
P4	5	256	32 x 32	16 x 16	1024 x 32 x 1024	256 x 32
P5	6	1024	16 x 16	32 x 32	256 x 16 x 256	1024 x 16
P6	7	4096	8 x 8	64 x 64	64 x 8 x 64	4096 x 8
P7	8	16384	4 x 4	128 x 128	16 x 4 x 16	16384 x 4
P8	9	65536	2 x 2	256 x 256	4 x 2 x 4	65536 x 2
P9	10	262144	1 x 1	512 x 512	1 x 1 x 1	262144 x 1

3.3. Apprentissage d'un réseau multicouche

Rappelons que le mode d'apprentissage du réseau de neurones sert à généraliser les connaissances du domaine de l'étude (une généralisation des connaissances). En d'autres termes, le réseau de neurones est capable d'accumuler l'information essentielle du domaine de l'étude, en éliminant l'information non essentielle ou redondante. Donc, nous pouvons restaurer l'information en utilisant un volume d'information moindre que pour l'information originale.

Nous avons appliqué une méthode d'apprentissage adaptative dirigée. Cette méthode assure une réception des coefficients de la transformation du réseau de neurones (des poids et des seuils des neurones du RN), en fonction du contenu de l'image.

L'apprentissage d'un réseau multicouche en mode dirigé est fondé sur une procédure de modification des poids synaptiques des couches cachées de fonction d'erreurs constatées entre la sortie désirée et la sortie réelle du réseau, quand on lui présente des observations prototypes étiquetées (Jolion, 2001). L'étape d'apprentissage du réseau de neurones est présentée sur la figure 3.6.

Nous travaillons avec une fonction d'entraînement qui actualise le poids selon l'algorithme de rétrocouplage élastique (Touzet, 1992) (voir l'annexe 5, page 4). La fonction d'activation *sigmoïde* a été choisie en vertu de sa capacité à répondre aux besoins de notre recherche (elle doit être dérivée pour calculer le gradient).

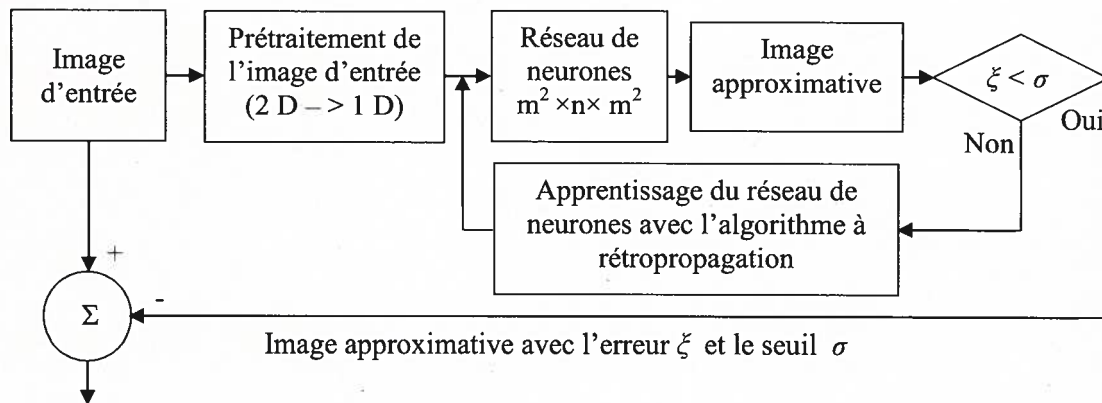


Figure 3.6 – Étape d'apprentissage du réseau de neurones

Le temps d'apprentissage du réseau des neurones joue un rôle essentiel pour les applications en temps réel. Nous divisons le nombre total des vecteurs d'entrées (base d'apprentissage) en trois parties.

Pendant l'entraînement dirigé, les données d'entrées et les données prévues (souhaitées) sont aléatoirement divisées en trois groupes : entraînement, épreuve et données de validation. Le réseau est entraîné sur les données d'entraînement, jusqu'à ce que ses performances commencent à stagner selon les données de validation, ce qui indique que la généralisation a atteint son optimum. Les données d'essai de la validation fournissent ainsi une épreuve complètement indépendante de la généralisation du réseau (Figure 3.7).

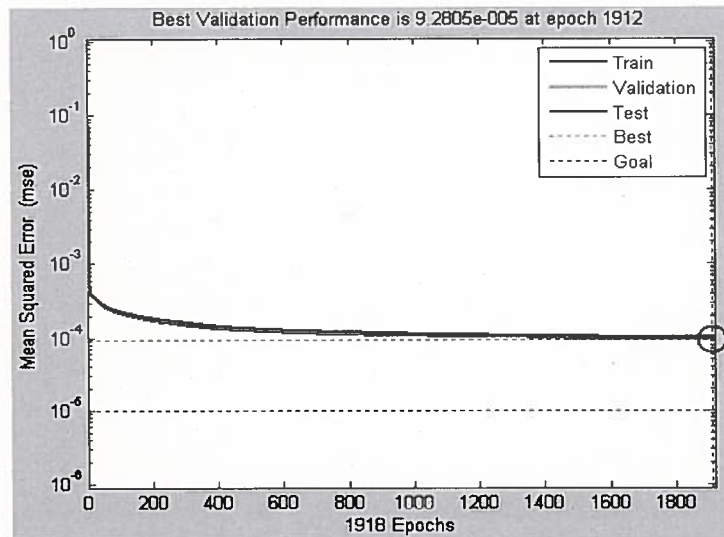


Figure 3.7 – Processus d'apprentissage du BPNN

La répartition générale est recommandée : 60 % - 80 % du nombre total des vecteurs d'entrée pour l'entraînement, c.-à-d. la plus grande partie; 10 % - 20 % sert à la validation; 10 % - 20 % sert à la mise à l'essai. Dans notre cas, le rapport choisi pour les vecteurs de contribution est : 60 % pour l'entraînement; 20 % pour la validation et 20 % pour la mise à l'essai (Figure 3.7).

3.4 Étapes importantes de la méthode proposée

Il y a trois étapes importantes dans l'approche proposée : l'entraînement, le codage et le décodage (Figure 3.8).

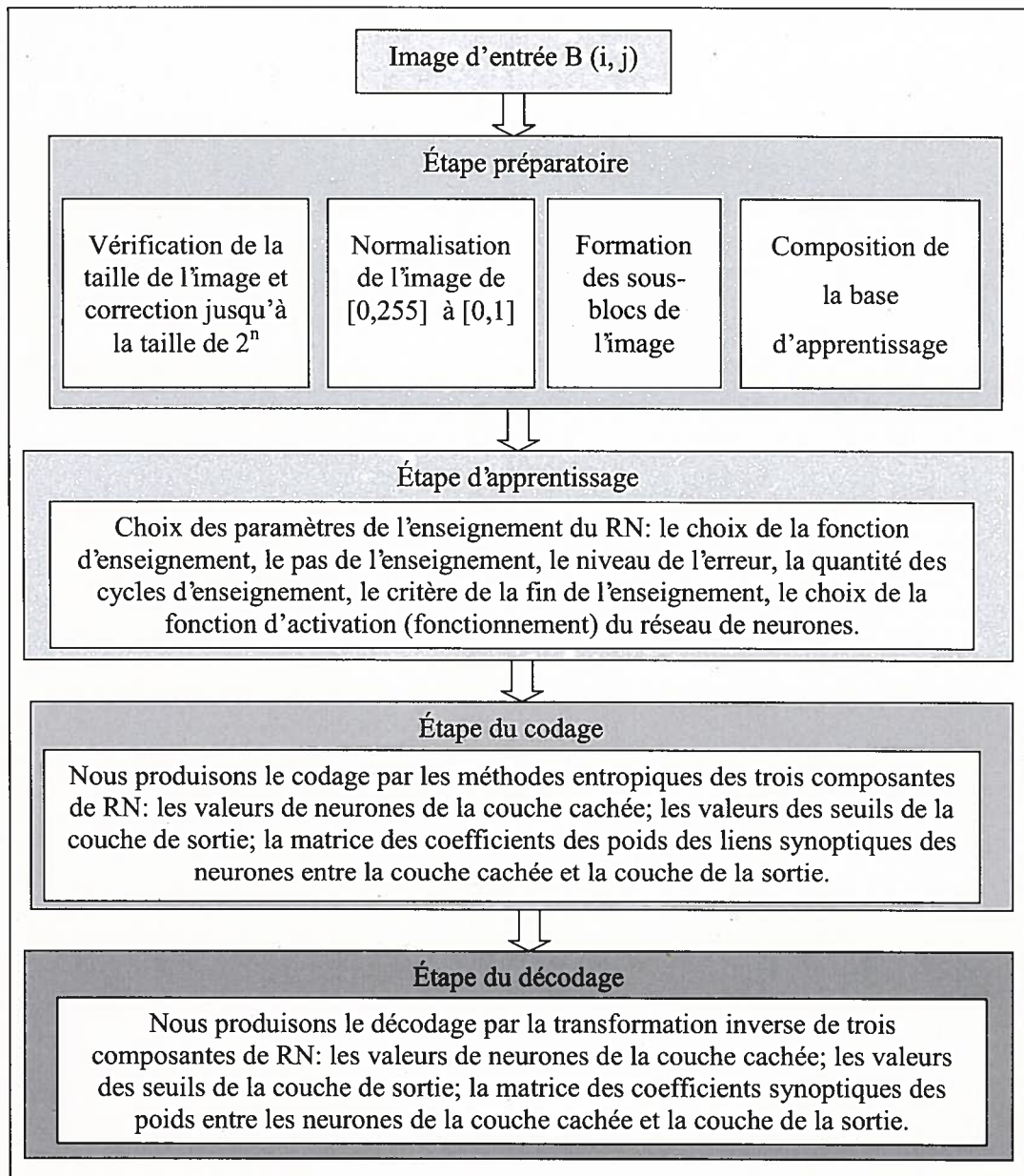


Figure 3.8 – Étapes de la méthode proposée

3.4.1. Entraînement du réseau de neurones

L'entraînement du réseau de neurones est suivi par les étapes suivantes :

- Créer un réseau de neurones avec :
- m^2 neurones de la couche d'entrée;

- n neurones de la couche cachée (le CR est le rapport de contribution des neurones cachés);
- m^2 neurones de la couche de la sortie.
- Diviser l'image d'entraînement en sous-blocs.
- Normaliser chaque bloc avec un nombre entier avec une valeur maximum de 255 (la valeur de luminance du pixel, entre 0 et 255) à un nombre réel avec une valeur maximum de 1 (entre 0 et 1).
- Utiliser les sous-blocs normalisés comme entrée au réseau de neurones.
- Régler les poids pour minimiser la différence entre l'entrée et la sortie (l'apprentissage du RN).

Répéter jusqu'à ce que l'erreur d'entraînement soit assez petite (moins ou égale au seuil) ($\xi \leq \sigma$).

3.4.2. Codage

- Diviser l'image en sous-blocs.
- Graduer chaque sous-bloc avec un nombre entier avec une valeur maximum de 255 à un nombre réel avec une valeur maximum de 1 (une normalisation des données).
- Utiliser les blocs normalisés comme l'entrée au réseau de neurones.
- Prendre les valeurs de la matrice des coefficients.
- Quantifier les données à 8 bits.
- Sauver les données quantifiées.

3.4.3. Décodage

- Lire dans un fichier des données quantifiées.
- Initialiser des éléments du réseau de neurones par des coefficients égaux aux données lues d'un fichier.
- Trouver la sortie de la couche de sortie.
- Restaurer une sortie réelle à une valeur de nombre entier entre 0 et 255.
- Mettre ces données dans l'image.
- Répéter pour chaque sous-bloc de l'image.

3.5. Réalisation algorithmique de l'approche hybride proposée

Les étapes suivantes ont été réalisées afin de développer l'algorithme.

a. L'image d'entrée $\mathbf{B}(i, j)$ avec la taille $V * H$ est divisée en K sous-images avec la taille de $m * m$ pixels. On a alors $k_0 = K$ sous-images au niveau $P = 0$ de la pyramide tronquée. (Figure 3.3 a). Nous avons l'image d'entrée de taille 1024 x 1024. Elle a été divisée en 4096 sous-images de la taille 16 x 16 chacune.

b. L'image originale est présentée sous forme d'une matrice (à deux dimensions, de taille 1024 x 1024). Nous l'avons divisée en 4096 sous-images (16 x 16). Ensuite, nous convertissons chaque sous-image à deux dimensions (matrice, 16 x 16) dans une dimension (en vecteur, 1 x 256). Pour cela, nous numérisons chacune des K sous-images séparément de gauche à droite, ligne après ligne et de haut en bas. Ainsi, pour le niveau $P = 0$ on obtient le vecteur d'entrée $I_{k_0} (k_0 = 1 \dots K)$ (4096 vecteurs avec 256 valeurs chacun).

c. Pour chaque 4096 sous-image de taille 16 * 16 pixels du niveau $P = 0$ de la pyramide, on utilise le même RN à trois couches (la couche d'entrée, la couche cachée et la couche de sortie). La couche d'entrée et la couche sortie comprennent $m^2 = 256$ neurones, la couche cachée comprend $m = 16$ neurones.

d. Pour l'apprentissage du réseau de neurones de niveau $P = 0$, nous utilisons tous les vecteurs d'entrée $I_{k_0}, (k = 1 \dots K)$ (la matrice d'entrée est 4096 x 256). Après l'étape d'apprentissage du réseau de neurones, nous recevons une matrice de coefficients de transformation (qui sont les valeurs de neurones de la couche cachée, toujours avec des valeurs de liaisons entre la couche cachée et celle de sortie et, finalement, des valeurs des seuils de la couche sortie).

e. On poursuit le traitement pour le niveau $P = 1$ (Figure. 3.9) où la taille des sous-images est égale à $m/2 * m/2$ et la quantité des sous-images augmente jusqu'à $k_1 = 4K$. Puis, nous continuons pour le niveau $P = 2$, en respectant les changements de la structure du réseau de neurones et de la taille de sous-images pour ce niveau de la pyramide.

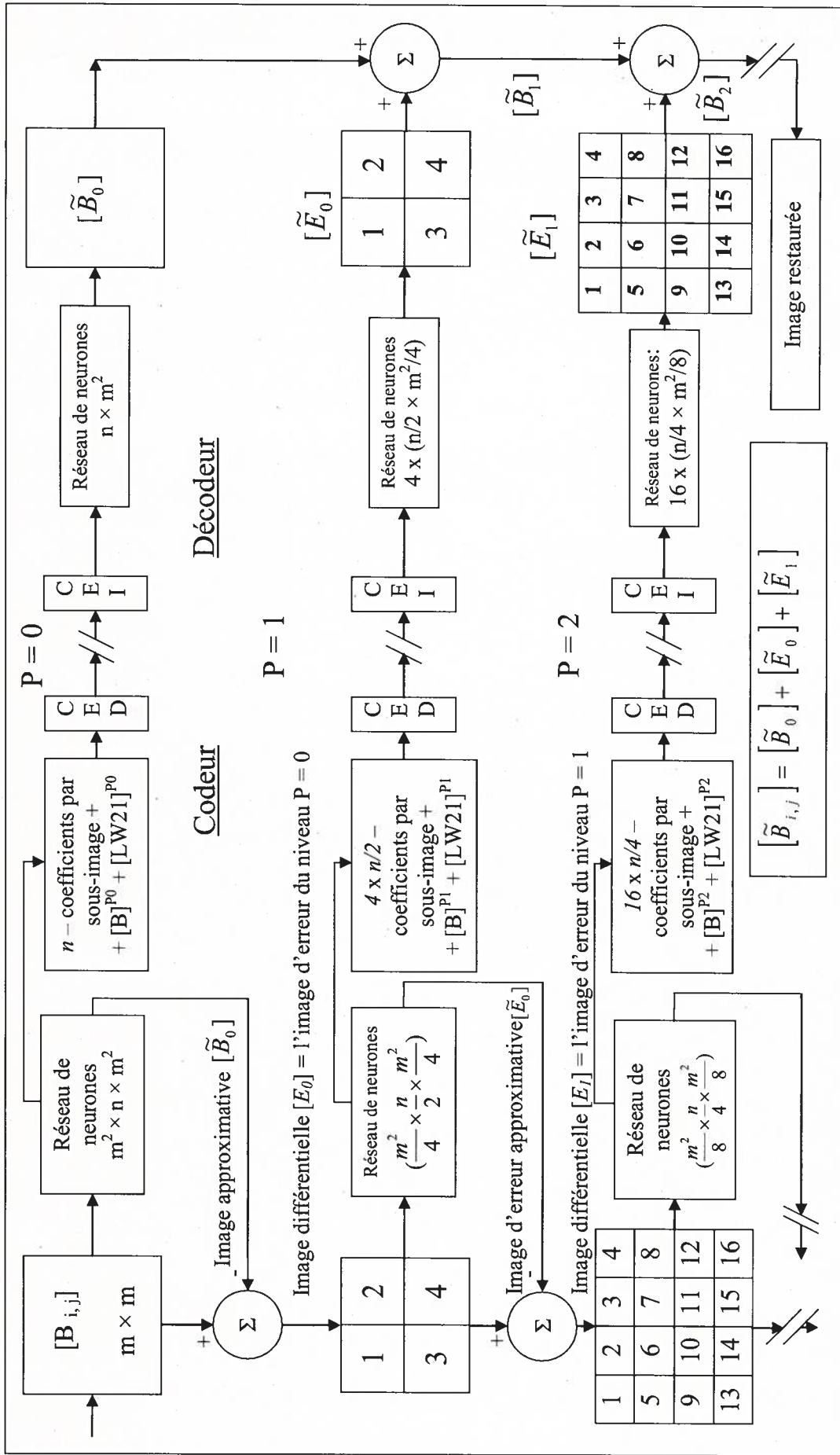


Figure 3.9 – Méthode de la décomposition pyramidale pour une image de taille $m \times m$ par des réseaux de neurones et des méthodes d'entropies du codage de l'information (CED – le codage d'entropie direct; CEI – le codage d'entropie inverse)

Et finalement, nous conservons les matrices de coefficients de transformation pour les niveaux de la pyramide correspondants.

g. Dans le décodeur, nous restituerons l'image en utilisant les données des matrices des coefficients de transformation pour des niveaux correspondants de la pyramide.

La figure 3.9 présente toutes ces étapes schématiquement.

3.6. Données des images utilisées

Premièrement, notre technique a été appliquée à l'image Ikonos de la ville de Sherbrooke, Québec, Canada. Les bandes visibles R.V.B. (4 m) ont été fusionnées avec la bande panchromatique (1 m) d'une image acquise le 19 juillet 2006 (GeoEye, 2006).

L'image originale de taille 8192x8192 a été divisée en 64 sous-images d'une taille de 1024x1024 chacune. Pour l'expérimentation, nous avons décidé de travailler avec 12 images de grandeur 1024x1024, 8 *bpp* (*bit par pixel*) (c'est-à-dire 1048576 octets chacune). Ces 12 images reflètent des zones essentielles de l'image originale. Ce sont des zones urbaines, des zones forestières, de l'eau, des montagnes, de l'autoroute et aussi des zones mélangées.

Deuxièmement, nous avons effectué nos recherches sur des images forestières de taille 224x352, 8 *bpp* (c'est-à-dire 78 848 octets chacune). Cette base de données d'images a été gracieusement présentée par le « Laboratoire de Traitement de Vidéo » du Département de radiocommunication et des technologies vidéo de l'Université Technique de Sofia, titulaire prof. R. Kountchev.

Le chapitre 4 présente, à travers trois articles, les résultats de la recherche.

4. Résultats

Les résultats de la recherche, les méthodes d'analyse de données et la discussion sur les résultats ont été publiés dans des revues à comité de lecture et sont intégrés dans le présent chapitre :

- 4.1 - Codage d'une série des images de télédétection;
 - 4.2 - Codage d'images de télédétection de THRS;
 - 4.3 - Transformation progressive d'images de télédétection en utilisant trois couches de la présentation pyramidale.
-
- 4.1 Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "Image decomposition on the basis of an inverse pyramid with 3-layer neural networks", *Journal of Communication and Computer*, ISSN 1548-7709, USA, Nov. 2009, Vol. 6, No.11 (Serial No.60), pp. 21-29
 - 4.2 Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "Compression of high-resolution satellite images with pyramidal neural network", International Science Press, *International Journal of Neural Networks and Applications*, 2(1) January-June 2009, pp. 5-14
 - 4.3 Valeriy Cherkashyn, Dong-Chen He and Roumen Kountchev, "A Novel Adaptive Representation Method (AIPR/BPNN) of Satellite Images Very High Definition", *Journal of Communication and Computer*, ISSN 1548-7709, USA, Sep. 2010, Vol. 7, No. 9 (Serial No. 70), pp. 55-66

Il est à noter que les 3 articles découlant de cette thèse présentent des éléments d'introduction et de méthodologie qui peuvent paraître redondants. Ces répétitions étaient nécessaires, afin de répondre aux exigences des réviseurs des journaux qui demandaient des explications sur la nouvelle méthode utilisée, sans se référer aux articles publiés précédemment. Cette approche permet aussi de garder une meilleure entité pour chacun des articles.

Décrivons brièvement le contenu de ces trois articles.

Le premier article présente un traitement d'une série d'images forestières de taille 224x352 pixels chacune par réseau de neurones à rétropropagation. L'image originale de taille 224x352 pixels était divisée en 1232 sous-images avec la taille 8x8 pixels. L'architecture du réseau de neurones compte 64 neurones pour la couche d'entrée et pour la couche de sortie et 8 neurones pour la couche cachée. Premièrement, nous avons testé 18 images avec 18 réseaux de neurones, donc nous avons utilisé une série des réseaux de neurones pour une série des images. Nous avons reçu le CR moyenne $CR=52,13$ (Tableau 1, page 71). Deuxièmement, nous avons utilisé un seul réseau de neurones pour toutes ces images. Le CR dans ce cas a fait 63,21 (Tableau 2, page73). Nos résultats montrent une amélioration du $CR=11,08$.

Cet article montre l'efficacité d'utilisation d'un réseau de neurones unique pour la série d'images, en comparaison avec l'utilisation d'une série de réseaux de neurones pour traiter chacune de ces 18 images séparément.

Le deuxième article traite de l'utilisation d'une méthode vers l'image de THRS de taille 8192x8192 pixels. Pour éliminer la complexité et le temps de traitement par logiciel tel que MATLAB, nous avons choisi 12 images qui représentent des zones représentatives de l'image originale. Chaque image était séparée en 4096 sous-images de taille 16x16 pixels. L'architecture du réseau de neurones compte 256 neurones pour la couche d'entrée et la couche de sortie et 16 neurones pour la couche cachée. Le codage a été effectué pour les trois différents niveaux de compression CR (121; 159; 273). Les résultats ont été comparés avec des méthodes de compression courantes telles que JPEG2000 et Lura Wave format.

La qualité visuelle des images restaurées par la nouvelle méthode est comparable aux méthodes courantes pour le niveau de compression $CR=107$. Pour le niveau de compression $CR=274$, nous avons obtenu une meilleure qualité visuelle pour les images compressées par notre méthode, en comparaison des méthodes courantes. Également, nous discutons de trois parties essentielles du réseau de neurones (des valeurs de neurones de la couche cachée, des poids synaptiques entre la couche

cachée et la couche de sortie et des coefficients des seuils des neurones de la couche de sortie), qui touchent l'efficacité de l'étape d'entraînement du RN.

Le troisième article traite du problème de la transformation progressive des images de THRS via les canaux de transmission. L'idée principale consiste à présenter l'image rapidement, mais à donner la possibilité à l'utilisateur de voir l'image de façon générale, pour chercher des points ciblés. En choisissant le point ciblé, nous procédons à l'amélioration de la zone de l'image choisie en ajoutant l'information de la couche suivante de la représentation pyramidale inverse.

Pour le niveau de la pyramide $P=0$, nous divisons l'image d'entrée de taille 1024×1024 pixels en 4096 sous-images. L'architecture du réseau de neurones compte 256 neurones pour la couche d'entrée et pour la couche de sortie et 16 neurones pour la couche cachée.

Pour le niveau de la pyramide $P=1$, nous divisons l'image d'erreur en 65536 sous-images de taille 8×8 pixels. L'architecture du réseau de neurones compte 64 neurones pour la couche d'entrée et pour la couche de sortie et 8 neurones pour la couche cachée. Nous avons aussi appliqué un nouveau critère d'évaluation SSIM de la qualité visuelle.

Nous avons réussi une amélioration de la qualité d'image restaurée de 4dB pour chaque couche additionnelle. Nous comparons la qualité de l'image restaurée avec des critères quantitatifs et qualitatifs.

**Image decomposition on the basis of an inverse pyramid
with 3-layer neural networks**

by

Valeriy Cherkashyn, Dong-Chen He
and Roumen Kountchev

Journal of Communication and Computer,

Volume 6, No.11 (Serial No.60)

ISSN 1548-7709, USA, Nov. 2009, pp. 21-29

Image decomposition on the basis of an inverse pyramid with 3-layer neural networks*

Valeriy Victorovich Cherkashyn¹, HE Dong-chen¹, Roumen Kirilov Kountchev²

(1. Centre for Research and Applications in Remote Sensing-CARTEL, University of Sherbrooke, Quebec J1K 2R1, Canada;

2. Department of Radio Communication and Video Technologies, Technical University-Sofia, Sofia 1000, Bulgaria)

Abstract: The contemporary information technologies and Internet impose high requirements on the image compression efficiency. Great number of methods for information redundancy reduction had already been developed, which are based on the image processing in the spatial or spectrum domain. Other methods for image compression use some kinds of neural networks. In spite of their potentialities, the methods from the last group do not offer high compression efficiency. New adaptive method for image decomposition on the basis of an inverse pyramid with neural networks is presented in this paper. The processed image is divided in blocks and then each is compressed in the space of the hidden layers of 3-layer BPNNs, which build the so-called inverse difference pyramid. The results of the new method modeling are presented for sequence of static images in comparison with results for single images from the same group.

Key words: pyramidal decomposition; image representation; image compression; neural networks

1. Introduction

The demands towards the efficiency of the methods for image representation in compressed form are getting higher together with their wide application

* **Acknowledgements:** This paper was supported by the NSERC (Natural Sciences and Engineering Research Council of Canada) to D.-C. He., CARTEL (Centre for Research and Applications in Remote Sensing), University of Sherbrooke, Quebec, Canada and National Fund for Scientific Research of Bulgarian Ministry of Education and Science, Contr. VU-I 305.

Corresponding author: Valeriy Victorovich Cherkashyn, Ph.D.; research fields: image processing, intellectual systems, GIS, artificial intelligence, neural networks, data processing, data mining, remote sensing.

HE Dong-chen, Ph.D., professor; research fields: image processing, classification, segmentation, mathematical morphology, GIS, remote sensing.

Roumen Kirilov Kountchev, Ph.D., D. Sc., professor; research fields: image processing, water marking, data processing, data mining, telecommunication.

and this is the basis for further elaboration and development. The classic methods^[1-6] could be classified in the following main groups: deterministic and statistical orthogonal linear transforms (DFT, DCT, WHT and KLT, SVD, PCA correspondingly); discrete wavelet transforms (DWT, Embedded Zerotree Wavelet, etc.); transforms based on prediction (LP, Adaptive LP, etc.); vector quantization (VQ, Adaptive VQ, Multistage Predictive VQ, etc.); fractal transforms (IFS, Quadtree Partitioned IFS, etc.) and decompositions, based on various pyramids, such as GP/LP, RLP, RSP/RDP, IDP, etc.

The analysis of these methods shows that for image compression are usually used deterministic orthogonal transforms and linear prediction with fixed coefficients^[1,4,6]. For example, in the standard JPEG is used discrete cosine transform (DCT) and linear prediction (LP), and the standard JPEG2000^[3] is based on the discrete wavelet transform (DWT). The transformed image data are compressed with entropy coding^[6], implemented as a combination of various methods for lossless coding (RLC, AC, LZW, etc.). The transforms DCT, DWT and LP are deterministic and the values of the coefficients of their matrices do not depend on the processed image content. For this reason, the compression efficiency is low when the correlation between the image content and the corresponding transform functions is low. The statistical transforms^[1,4] are more efficient than the deterministic ones, but they have higher computational complexity. Similar disadvantage have the fractal

methods^[2,6], which together with this are not enough efficient when images with unclear texture regions are processed. The famous pyramidal image decompositions are usually implemented with digital filters with fixed decimation and interpolation coefficients^[2,4] or use some kind of transform, such as for example the Multiple DCT^[5], i.e. these methods are not well conformed to the image content.

A group of methods for image representation, based on the use of artificial neural networks (NN)^[7-14] had recently been developed. Unlike the classic methods, this approach is distinguished by higher compression ratios, because together with the coding, NN training is performed. The results already obtained show that these methods can not successfully compete the still image compression standards, JPEG and JPEG2000^[3]. For example, the Adaptive Vector Quantization (AVQ), based on SOM NN^[8,13], requires the use of code books of too many vectors, needed to ensure high quality of the restored image and this results in lower compression. One of the first encouraging results for a similar method of research have been received in previously article^[15].

In this paper is offered new adaptive method for inverse pyramidal decomposition of digital images with 3-layer BPNNs. The results obtained with the method modeling show significant visual quality enhancement for sequence of static images in comparison with results for single images from the same group.

The paper is arranged as follows: In Section 2 is described the method for adaptive pyramidal image representation; in Section 3 is given the algorithm simulation; in Section 4 are given some experimental results, and Section 5 is the Conclusion.

2. Method for adaptive pyramidal image representation

2.1 Pyramidal decomposition selection

The basic advantage of the pyramidal decomposition in comparison with the other methods for image compression is the ability to perform “progressive” transfer (or storage) for every consecutive decomposition layer. In result, the image could be restored with high compression ratio and gradually improving quality. The classic approach for progressive image transfer is based on the Laplasian pyramid (LP)^[4] combined with the Gaussian (GP).

In this paper is offered new approach for pyramidal image representation, based on the so-called *Adaptive Inverse Difference Pyramid* (AIDP). Unlike the non-adaptive Inverse Difference Pyramid (IDP)^[5] it is built in the non-linear transformed image space using a group of NNs. The AIDP is calculated starting the calculation from the pyramid top, placed down, and continues iteratively with the next pyramid layers.

The AIDP built this way, has some important advantages when compared to the LP: easier implementation of the progressive image transfer and compression enhancement in the space of the hidden layers of the corresponding NN.

2.2 Description of the inverse difference pyramid

Mathematically the digital image is usually represented as a matrix of size $H \times V$, whose elements $b(x, y)$ correspond to the image pixels; x and y define the pixel position as a matrix row and column and the pixel brightness is b . The halftone image is then defined as:

$$[B(x,y)] = \begin{bmatrix} b(0,0) & b(0,1) & \dots & b(0,H-1) \\ b(1,0) & b(1,1) & \dots & b(1,H-1) \\ \vdots & \vdots & & \vdots \\ b(V-1,0) & b(V-1,1) & \dots & b(V-1,H-1) \end{bmatrix} \quad (1)$$

In order to make the calculation of the pyramidal image decomposition easier, the matrix is divided into K blocks (sub-images) of size $m \times n$ ($m=2^p$) and on each is then built a multi-layer IDP. The number p of the IDP layers for every block is in the range $0 \leq p \leq n-1$. The case $p=n-1$ corresponds to complete pyramidal decomposition of maximum number of layers, for

which the image is restored without errors (all decomposition components are used).

The IDP top (layer $p=0$) for a block of size $2^n \times 2^n$ contains coefficients, from which after inverse transform is obtained its worse (coarse) approximation. The next IDP layer for the same block (the layer $p=1$) is defined from the difference between the block matrix and the approximation, divided into 4 sub-matrices of size $2^{n-1} \times 2^{n-1}$ in advance. The highest IDP layer (layer $p=n-1$) is based on the information from the pixels in all the 4^{n-1} difference sub-matrices of size 2×2 , obtained in result of the $(n-1)$ -time division of the initial matrix into sub-matrices.

In correspondence with the described principle, the matrix $[B_{k_0}]$ of one image block could be represented as a decomposition of $(n+1)$ components:

$$[B_{k_0}] = [\tilde{B}_{k_0}] + \sum_{p=1}^{n-1} [\tilde{E}_{k_{p-1}}] + [E_{k_n}] \quad (2)$$

for $k_p=1, 2, \dots, 4^p K$ and $p=0, 1, \dots, n-1$.

Here k_p is the number of the sub-matrices of size $m_p \times m_p$ ($m_p=2^{n-p}$) in the IDP layer p ; the matrices $[\tilde{B}_{k_0}]$ and $[\tilde{E}_{k_{p-1}}]$ are the corresponding approximations of $[B_{k_0}]$ and $[E_{k_{p-1}}]$; $[E_{k_n}]$ is the matrix, which represents the decomposition error in correspondence with Eq. (2), for the case, when only the first n components are used.

The matrix $[E_{k_{p-1}}]$ of the difference sub-block k_{p-1} in the IDP layer p is defined as:

$$[E_{k_{p-1}}] = [E_{k_{p-2}}] - [\tilde{E}_{k_{p-2}}], \quad (3)$$

for $p=2, 3, \dots, n-1$. In this case $p=1$:

$$[E_{k_0}] = [B_{k_0}] - [\tilde{B}_{k_0}] \quad (4)$$

The matrix $[E_{k_{p-1}}]$ of the difference sub-block in the layer p is divided into $4^p K$ sub-matrices $[E_{k_p}]$ and for each is then calculated the corresponding approximating matrix $[\tilde{E}_{k_p}]$. The submatrices $[\tilde{E}_{k_p}]$ for $k_p=1, 2, \dots, 4^p K$ define the next decomposition component $(p+1)$, represented by Eq. (2). For this is

necessary to calculate the new difference matrix and then to perform the same operations again following the already presented order.

2.3 Image representation with AIDP-BPNN

The new method for image representation is based on the IDP decomposition, in which the direct and inverse transforms in all layers are performed using 3-layer neural networks with error back propagation (BPNN)^[7].

The general BPNN structure in AIDP was chosen to be a 3-layer one of the kind $m^2 \times n \times m^2$, shown in Fig.1. The input layer is of m^2 elements, which correspond to the input vector components; the hidden layer is of n elements for $n < m^2$, and the output layer is of m^2 elements as well, which correspond to the output vector components. The input m^2 -dimensional vector is obtained in result of the transformation of the elements m_{ij} of each image block of size $m \times m$ into one-dimensional massif of length m^2 using the "meander" scan, shown in Fig. 2.

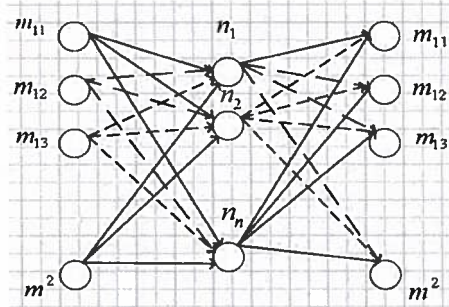


Fig. 1 Three-layer BPNN with $n < m^2$ neurons in the hidden layer and m^2 neurons in the input and in the output layer

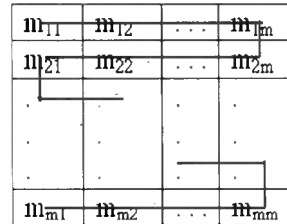


Fig. 2 The "meander" scan applied on image block of size $m \times m$

In order to obtain better compression the processed image is represented by the sequence of m^2 -dimensional vectors $\vec{X}_1, \vec{X}_2, \dots, \vec{X}_K$, which are then transformed in the n -dimensional vectors $\vec{h}_1, \vec{h}_2, \dots, \vec{h}_K$ correspondingly. The components of the vectors \vec{h}_k for $k=1, 2, \dots, K$ represent the neurons in the hidden layer of the trained 3-layer BPNN with $m^2 \times n \times m^2$ structure. In the output NN layer the vector \vec{h}_k is transformed back into the m^2 -dimensional output vector \vec{Y}_k , which approximates the corresponding input vector \vec{X}_k . The approximation error depends on the training algorithm and on the participating BPNN parameters. The training vectors $\vec{X}_1, \vec{X}_2, \dots, \vec{X}_K$ at the BPNN input for the AIDP layer $p=0$ correspond to the image blocks. For the training was chosen the algorithm of Levenberg-Marquardt (LM)^[7,8], which ensures rapidity in cases, when high accuracy is not required and as a result is suitable for the presented approach. One more reason is that the data necessary for the training has significant volume and information redundancy, but this does not make worse the training with the LM algorithm and influences only the time needed (i.e. it becomes longer).

The parameters of the 3-layer BPNN define the relations between the inputs and the neurons in the hidden layer, and between the neurons from the hidden and the output layer. These relations are described using weight matrices and vectors, which contain threshold coefficients, and with functions for non-linear vector transform.

The relation between the input m^2 -dimensional vector \vec{X}_k and the corresponding n -dimensional vector \vec{h}_k in the hidden BPNN layer for the AIDP layer $p=0$ is:

$$\vec{h}_k = f([W]_1 \vec{X}_k + \vec{b}_1) \quad \text{for } k=1, 2, \dots, K, \quad (5)$$

where $[W]_1$ is the matrix of the weight coefficients of size $m^2 \times n$, which is used for the linear transform of the input vector \vec{X}_k ; \vec{b}_1 is the n -dimensional vector of the threshold coefficients in the hidden layer, and $f(x)$ is

a linear activating sigmoid function, defined by the relation:

$$f(x) = 1/(1 + e^{-x}) \quad (6)$$

In result the network performance becomes partially non-linear and this influence is stronger when x is outside the range $[0, +1]$.

The relation between the n -dimensional vector \vec{h}_k of the hidden layer and the m^2 -dimensional BPNN vector \vec{Y}_k from the AIDP layer $p=0$, which approximates \vec{X}_k , is defined in accordance with Eq. (5) as follows:

$$\vec{Y}_k = f([W]_2 \vec{h}_k + \vec{b}_2) \quad \text{for } k=1, 2, \dots, K, \quad (7)$$

where $[W]_2$ is a matrix of size $n \times m^2$ representing the weight coefficients used for the linear transform in the hidden layer of the vector \vec{h}_k , and \vec{b}_2 is the m^2 -dimensional vector of the threshold coefficients for the output layer. Unlike the pixels in the halftone images, whose brightness is in the range $[0, 255]$, the components of the input and output BPNN vectors are normalized in the range $x_i(k), y_i(k) \in [0, 1]$ for $i=1, 2, \dots, m^2$.

The components of the vector which represents the neurons in the hidden layer $h_j(k) \in [0, 1]$ for $j=1, 2, \dots, n$ are placed in the same range, because they are defined by the activating function $f(x) \in [0, 1]$. The normalization is necessary, because it enhances the BPNN efficiency^[8].

The adaptive image representation with AIDP-BPNN method is performed in two consecutive stages:

- (1) BPNN training,
- (2) coding of the obtained output data.

For the BPNN training in the AIDP layer $p=0$, the vectors \vec{X}_k are used as input and reference ones, with which are compared the corresponding output vectors. The comparison result is used to correct the weight and the threshold coefficients so that to obtain minimum MSE:

The training is repeated until the MSE value for the output vectors becomes lower than predefined threshold.

For the training of the 3-layer BPNN in the next ($p > 0$) AIDP layers are used the vectors obtained after the dividing of the difference block $[E_{k_{p-1}}]$ (or sub-block) into $4^p K$ sub-blocks and their transformation into corresponding vectors. The BPNN training for each layer $p > 0$ is performed in the way already described for the layer $p = 0$.

In the second stage the vectors in the hidden BPNN layers for all AIDP layers are coded losslessly with entropy coding. The coding is based on two methods: Run-Length Coding (RLC) and variable length Huffman coding^[6].

The block diagram of the pyramid decomposition for one block of size $m \times m$ with 3-layer BPNN for the layers $p = 0, 1, 2$ and entropy coding/decoding is shown in Fig. 3. When the BPNN training is finished, for each

layer p are defined the corresponding output weight matrix $[W]_p$ and the threshold vector $[b]_p$.

The entropy coder (EC) compresses the data transferred to the decoder for the layer p , i.e.:

The vector of the threshold coefficients for the neurons in the output NN layer (common for all blocks in the layer p);

The matrix of the weight coefficients of the relations between the neurons in the hidden layer towards the output BPNN layer (common for all blocks in the layer p);

The vector of the neurons in the hidden BPNN layer is personal for each block in the layer p .

In the decoder is performed the entropy decoding (ED) of the compressed data. After that the BPNN in the layer p is initialized setting the values of the threshold coefficients for the neurons in the output layer and of the weight coefficients for the neurons, connecting the hidden and the output layers.

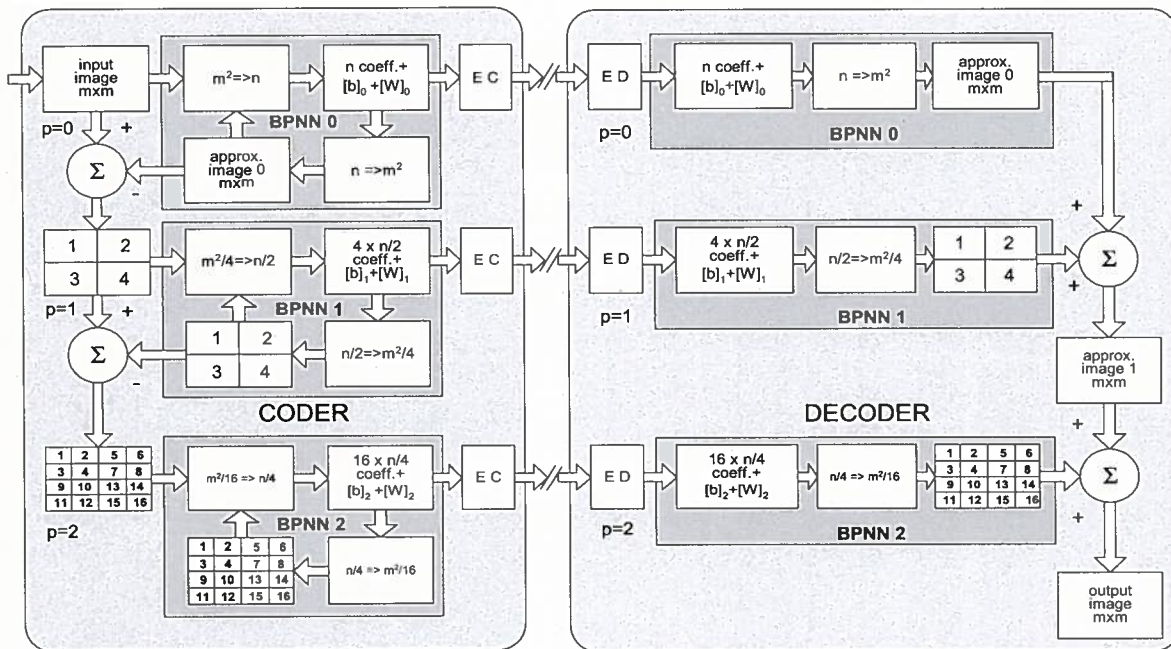


Fig. 3 Block diagram of the 3-layer inverse pyramidal image decomposition with 3-layer BPNN

Note: Here $[b]_p$ – the vector of the threshold coefficients in the output layer for $p = 0, 1, 2$; $[W]_p$ – the matrix of the weight coefficients between the hidden and the output BPNN layer for $p = 0, 1, 2$.

At the end of the decoding the vector of the neurons in the hidden BPNN layer for each block is transformed into corresponding output vector. The obtained output vectors are used for the restoration of the processed image.

3. Simulation of the AIDP-BPNN algorithm

For the simulation of the AIDP-BPNN algorithm is necessary to perform the following operations:

- (1) transformation of the input data into a sequence of vectors;
- (2) selection of the BPNN structure;
- (3) BPNN creation and initialization of its parameters;
- (4) BPNN training using the input vectors so that to obtain the required output vectors;
- (5) testing of the AIDP-BPNN algorithm with various test images and evaluation of their quality after restoration (objective and subjective).

The AIDP-BPNN algorithm consists of the following four basic steps (for more details^[16]):

- (1) preliminary preparation of entrance data;

- (2) training of neural network;
- (3) coding;
- (4) decoding.

4. Experimental results

The experiments with the AIDP-BPNN algorithm were performed with test images of size 224×352 , 8 *bpp* (i.e. 78 848 B), and original images are presented in Fig. 4. In the ADP layer $p=0$ the image is divided into K blocks of size 8×8 pixels, ($K=1232$). At the BPNN input for the layer $p=0$ is passed the training matrix of the input vectors of size $64 \times 1232=78\ 848$. In the hidden BPNN layer the size of each input vector is reduced from 64 to 8.

The restoration of the output vector in the decoder is performed using these 8 components, together with the vector of the threshold values and the matrix of the weight coefficients in the BPNN output layer. For the layer $p=0$ the size of the data obtained is 83 456 B, i.e. - larger than that of the original image (78 848 B). As it was already pointed out, the data has high correlation and is efficiently compressed with entropy coding.

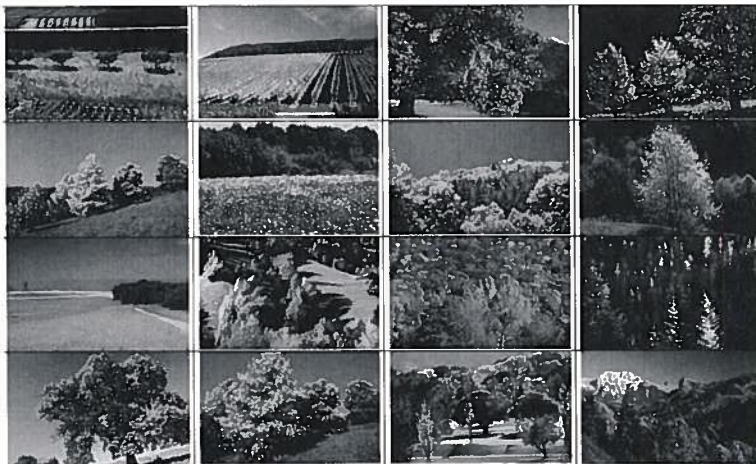


Fig. 4 Original Test Images "forest"

For example, the compressed data size for the same layer ($p=0$) of the test image *Grayscale_forest010032.bmp* is 1510 B (the result is

given in Table 1). Taking into account the size of the original image, is calculated the compression ratio $Cr=52.21$.

The Peak Signal to Noise Ratio for the first test image *Grayscale_forest010032.bmp* for $p=0$ (Table 1) is $PSNR=23.45$ dB. In the same table are given the compression ratios obtained with AIDP-BPNN for other 18 test images of same size (224×352). It is easy

to see that for the mean compression ratio $Cr=52.13$ is obtained $PSNR>22.52$ dB, i.e. the visual quality of the restored test images (Fig. 5) is good enough for various applications.

Table 1 Results obtained for 18 test images “forest” after AIDP-BPNN compression, 78 848 Bytes for each image

File name	Cr	$PSNR$ [dB]	$RMSE$	Bit-rate/pixel (bpp)	Compression file (B)
1	52.21	23.45	17.36	0.1532	1510
2	52.42	23.05	17.94	0.1526	1504
3	51.53	19.10	28.27	0.1552	1530
4	50.54	17.14	35.45	0.1583	1560
5	52.35	22.71	18.68	0.1528	1506
6	52.35	19.72	26.32	0.1528	1506
7	53.06	22.28	19.61	0.1508	1486
8	52.49	23.4	17.25	0.1524	1502
9	52.85	31.63	06.69	0.1514	1492
10	51.80	21.55	21.33	0.1544	1522
11	52.21	21.92	20.45	0.1532	1510
12	52.21	22.28	19.62	0.1532	1510
13	52.35	19.87	25.88	0.1528	1506
14	51.73	19.50	27.00	0.1546	1524
15	52.43	22.31	19.55	0.1526	1504
16	52.08	23.67	16.71	0.1536	1514
17	52.49	28.07	10.07	0.1524	1502
18	51.20	23.63	16.80	0.1563	1540
The average result for all 18 images:					
	52.13	22.52	20.28	0.1535	1513

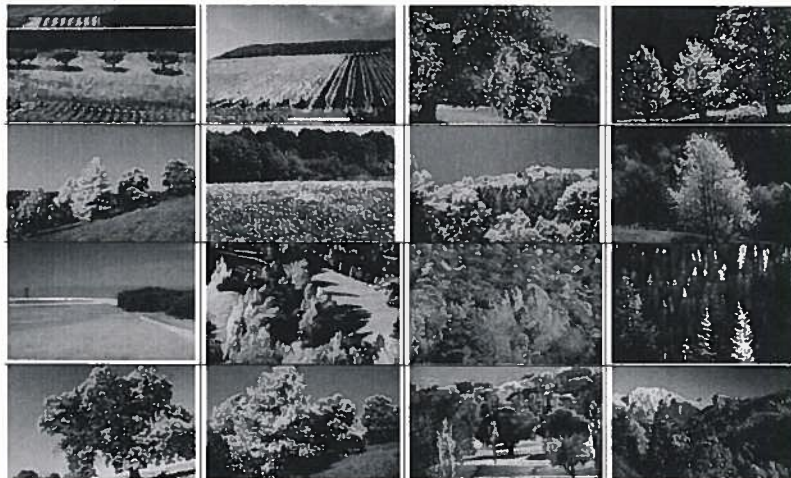


Fig. 5 Restored test image “forest” by AIDP-BPNN method

Table 2 Results obtained for sequence of 18 test images “forest” after AIDP-BPNN compression; 1419264 B for this consequence (Fig. 6)

File name	C_r	$PSNR$ [dB]	$RMSE$	Bit-rate/ pixel (bpp)	Compression file (B)
all	63.21	21.35	17.36	0.1266	22452

In Table 2 are given the results for the sequence of 18 test images “forest” after AIDP-BPNN compression, implemented with MATLAB.



Fig. 6 Original sequence of test images, 224x352x18 = 1419264 B



Fig. 7 Restored sequence of test images “forest” by AIDP-BPNN method

Thus, the results received show higher degree of compression $C_r=63.21$ for sequence of images in comparison with average value of factor of compression $C_r=52.13$ for each image separately. The gain is on 11.08 times.

The NN architecture used for the experiments comprises for the zero level 64 neurons in the input layer, 8 neurons in the hidden layer, and 64 neurons in the output layer. The chosen proportion for the input vectors was correspondingly: 80% for *Training*; 10 % for *Validation* and 10% for *Testing*.

The sequence of static images presented on Fig. 6 makes the image in the size 224x352x18, that is 1 419 264 B.

5. Conclusions

In this paper is presented one new approach for still image adaptive pyramid decomposition based on the AIDP-BPNN algorithm. The algorithm modeling was used to compare it with version of the sequence of 18 test images “forest”. The results obtained show that for the sequence of 18 test images “forest” it ensures higher compression ratio of the restored sequence of test images.

The AIDP-BPNN is asymmetric (the coder is more complicated than the decoder) and this

The results obtained (Table 2) show that AIDP-BPNN method for sequence of test images surpasses the compared methods with fixed image (Table 1).

The restored sequence of test images “forest” is presented in Fig. 7

determines it mostly in application areas which do not require real time processing i.e. applications, for which the training time is not crucial.

The hardware implementation of the method is beyond the scope of this work. The experiments for the AIDP-BPNN algorithm were performed with sub-blocks of size 8x8 pixels. The computational complexity of the method depends on the training method selected.

The new method offers wide opportunities for application areas in the digital image processing, such as the progressive transfer via Internet, saving and searching in large image databases, remote sciences, etc.

A result received in the given work confirms theoretical pre-conditions, that the images of the high resolution and very high resolution (satellite images, multispectral images, hyperspectral images) can possess the increased degree of redundancy of data. This circumstance allows applying effectively method AIDP-BPNN to a compression of images of the high resolution in remote sciences.

References:

- [1] K. Rao, P. Yip. *The Transform and Data Compression Handbook*. CRC Press LLC, 2001.

- [2] M. Barni. *Document and Image Compression*. CRC Press Taylor and Francis Group, 2006.
- [3] T. Acharya, P. Tsai. *JPEG2000 Standard for Image Compression*. John Wiley and Sons, 2005.
- [4] R. Gonzales, R. Woods. *Digital Image Processing*. Prentice Hall, 2002.
- [5] R. Kountchev, V. Haese-Coat, J. Ronsin. Inverse pyramidal decomposition with multiple DCT. *Signal Processing: Image Communication, Elsevier 17*, 2002: 201-218.
- [6] D. Salomon. *Data Compression*. Springer, 2004.
- [7] St. Perry, H. Wong, L. Guan. *Adaptive Image Processing: A Computational Intelligence Perspective*. CRC Press LLC, 2002.
- [8] Y. H. Hu, J.N. Hwang (Eds.). *Handbook of Neural Network Signal Processing*. CRC Press LLC, 2002.
- [9] R. Dony, S. Haykin. Neural network approaches to image compression. *Proceedings of the IEEE*, 1995, 23(2): 289-303.
- [10] A. Namphol, *et al.* Image compression with a hierarchical neural network. *IEEE Trans. on Aerospace and Electronic Systems*, 1996, 32(1): 327-337.
- [11] S. Kulkarni, B. Verma, M. Blumenstein. Image compression using a direct solution method based neural network. *10-th Australian Joint Conference on AI, Perth, Australia*, 1997: 114-119.
- [12] J. Jiang. Image compressing with neural networks: A survey. *Signal Processing: Image Communication, Elsevier*, 1999, 14(9): 737-760.
- [13] N. Kouda, *et al.* Image compression by layered quantum neural networks. *Neural Processing Lett.*, 2002, 16: 67-80.
- [14] *International Journal on Graphics, Vision and Image Processing (GVIP)*, Special Issue on Image Compression, 2007.
- [15] V. Cherkashyn, N. Hikal, R. Kountchev, Y. Biletskiy. Image compression based on the inverse difference pyramid with BPNN. *proc. of IEEE International Symposium on Industrial Electronics (ISIE'06)*, 2006: 645-648.
- [16] V. Cherkashyn, R. Kountchev, D.-C. He, R. Kountcheva. Adaptive image pyramidal representation. *IEEE Symposium on Signal Processing and Information Technology, ISSPIT'08*.

(Edited by Jane, Sang)

(continued from Page 7)

- [35] J. Skorin-Kapov. Tabu search applied to the quadratic assignment problem. *ORSA Journal on Computing*, 1990, 2: 33-45.
- [36] É. D. Taillard. Robust taboo search for the quadratic assignment problem. *Parallel Computing*, 1991, 17: 443-455.
- [37] R. K. Ahuja, J. B. Orlin and A. Tiwari. A descent genetic algorithm for the quadratic assignment problem. *Computers and Operations Research*, 2000, 27: 917-934.
- [38] Z. Drezner. A new genetic algorithm for the quadratic assignment problem. *INFORMS Journal on Computing*, 2003, 115: 320-330.
- [39] W. P. Adams, H. D. Sherali. A tight linearization and an algorithm for zero-one quadratic programming problems. *Manage Sci*, 1986, 32(10): 1274-1290.
- [40] XIA Y. and Yuan Y. A new linearization method for quadratic assignment problems. *Optimization Methods and Software*, 2006, 21(5): 803-816.
- [41] K. M. Anstreicher. Recent advances in the solution of quadratic assignment problems. *Mathematical Programming Ser. B*, 2003, 97: 24-42.
- [42] P. M. Pardalos, F. Rendl and H. Wolkowicz. The quadratic assignment problem: A survey and recent developments in quadratic assignment and related problems. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, AMS, Rhode Island, 1994, 16: 1-42.
- [43] V. D. Cung, T. Mautor, P. Michelon and A. Tavares. A scatter search based approach for the quadratic assignment problem. In T. Bäck, Z. Michalewicz and YAO X. (Eds.), *Proceedings of IEEE-ICEC-EPS'97, IEEE International Conference on Evolutionary Computation and Evolutionary Programming Conference*, Indianapolis, 1997: 165-170.
- [44] B. A. Murtagh B.A., T. R. Jererson and V. Sornprasit. A heuristic procedure for solving the quadratic assignment problem. *European Journal of Operational Research*, 1982, 9: 71-76.

(Edited by Jane, Sang)

**Compression of high-resolution satellite images with
pyramidal neural network**

by

Valeriy Cherkashyn, Dong-Chen He
and Roumen Kountchev

International Science Press

International Journal of Neural Networks and Applications

2(1) January-June 2009, pp. 5-14

COMPRESSION OF HIGH-RESOLUTION SATELLITE IMAGES WITH PYRAMIDAL NEURAL NETWORK

¹Valeriy Cherkashyn & ²Dong-Chen He

CARTEL, Université de Sherbrooke Sherbrooke, QC, J1K 2R1 Canada
E-mails: ¹valeriy.cherkashyn@usherbrooke.ca, ²dong-chen.he@usherbrooke.ca

³Roumen Kountchev

Dept. of RCVT, Technical University-Sofia Kl. Ohridski 8, Sofia 1000 Bulgaria
E-mail: rkountch@tu-sofia.bg

Abstract: The fast development of the remote sensing systems investigating the earth surface settled high requirements on the representation of satellite images in compressed form, which to permit their restoration with high visual quality. Significant number of methods had already been developed aimed at the information redundancy reduction, most of them based on the image processing in the spatial or spectrum domain. Another significant group of image compression methods use artificial neural networks of various kinds. Despite their high potential the compression efficiency of the methods from the last group is still not satisfactory. In this paper is offered a new method for the representation of high-resolution satellite images in compressed form, using a pyramidal decomposition based on neural network with error backpropagation (BPNN). Each sub-block of the processed image is compressed in the space of the hidden layers of 3-layer BPNNs, which build the so-called Inverse Difference Pyramid. The results obtained from the new method modeling for compression of satellite images were compared with those obtained for the standards JPEG and JPEG2000

Keywords: Image representation and compression, Neural networks with error backpropagation, Pyramidal image decomposition

1. INTRODUCTION

Remote sensing operates with huge volumes of information. First of all it concerns high-resolution satellite images, multi-spectral and hyper-spectral images. In order to make the modern process of information gathering easier, the efficient storage and transmission of the related data demands a revision of the existing approaches for still image compression. The creation of new methods for image data representation aimed at the image size reduction is a problem of high importance. The compression is achieved by the removal of one or more of three basic redundancies [10]:

- (1) *coding redundancy*, which exists when less than optimal code words are used;
- (2) *interpixel redundancy*, which results from correlations between the pixels of an image;
- (3) *psychovisual redundancy*, which is due to data that is ignored by the human visual system (i.e., visually nonessential information).

The classic methods for still image compression [1-3, 9, 14, 11] could be classified in the following main groups:

- deterministic and statistical orthogonal linear transforms (DFT, DCT, WHT and KLT, SVD, PCA correspondingly);
- discrete wavelet transforms (DWT, Embedded Zerotree Wavelet, Enhanced Compression Wavelets – ECW, etc.);
- transforms based on linear prediction (LP, Adaptive LP, etc.);
- vector quantization (VQ, Adaptive VQ, Multistage Predictive VQ, etc.);
- fractal transforms (IFS, Quadtree Partitioned IFS, etc.);
- decompositions, based on various pyramids, such as GP/LP, RLP, RSP/RDP, IDP, etc.

The analysis of these methods shows that for image compression are usually used [5, 9, 3]:

- deterministic orthogonal transforms;
- linear prediction with fixed coefficients.

In the JPEG standard are used the discrete cosine transform (DCT) and linear prediction (LP), and the JPEG2000 standard [14] is based on the discrete wavelet transform (DWT). The transformed image data are compressed with entropy coding [3], implemented as a combination of various methods for lossless coding (RLC, AC, LZW, etc.).

The DCT, DWT and LP transforms are deterministic and the values of the coefficients of their matrices do not depend on the processed image content. For this reason, the compression efficiency is low when the correlation between the image content and the corresponding transform functions is low.

The statistical transforms (KLT, SVD, PCA) are more efficient than the deterministic ones [5, 9], but they have higher computational complexity. Similar disadvantage have the fractal methods [3, 6], which together with this are not enough efficient when images with unclear texture regions are processed.

The famous pyramidal image decompositions are usually implemented with digital filters with fixed decimation and interpolation coefficients [6, 9] or use some kind of transform, such as for example the Multiple DCT [11], i.e. these methods are not well conformed to the image content.

A group of methods for image representation, based on the use of artificial neural networks (NN) [1, 2, 4, 7, 8, 12, 13, 18] had recently been developed. Unlike the classic methods, this approach is distinguished by higher compression ratios, because together with the coding, NN training is performed. The results already obtained show that these methods can not successfully compete the still image compression standards, JPEG and JPEG2000 [14]. For example, the Adaptive Vector Quantization (AVQ), based on Self Organizing Map (SOM) NN [7, 18], requires the use of code books of too many vectors, needed to ensure high quality of the restored image and this results in lower compression.

In this paper is offered new method for inverse pyramidal decomposition of digital images with 3-layer Backpropagation NN (BPNN). The results obtained with the method modeling show significant visual quality enhancement for a group of similar static images in comparison with the results for single images from the same group.

The paper is arranged as follows: in Section II is described the method for Pyramidal Neural Network (PNN) image representation, in Section III are given some experimental results, and Section IV is the Conclusion.

2. METHOD FOR PNN IMAGE REPRESENTATION

2.1. Pyramidal Decomposition Selection

The basic advantage of the pyramidal decomposition in comparison with other methods for image compression is the ability to perform “progressive” transfer (or storage) of the approximating image obtained for every consecutive decomposition layer. In result, the image could be first restored with high compression ratio and relatively low quality and permits quality improvement on request.

The classic approach for progressive image transfer is based on the Laplasian pyramid (LP) [9] combined with the Gaussian (GP).

In this paper is offered one new approach for pyramidal image representation, based on the so-called *Adaptive Inverse Difference Pyramid* (AIDP). Unlike the non-adaptive Inverse Difference Pyramid (IDP) [11] it is built in the non-linear transformed image space using a group of NNs.

The AIDP is calculated starting the calculation from the pyramid top, placed down, and continues iteratively with the next pyramid layers.

Representation thus constructed, has some important advantages when compared to the LP: easier implementation of the progressive image transfer and compression enhancement in the space of the hidden layers of the corresponding NN.

2.2. Description of the Inverse Difference Pyramid (IDP)

Mathematically the digital image is usually represented as a matrix of size $H \times V$, whose elements $b(x, y)$ correspond to the image pixels; x and y define the pixel position as a matrix row and column and the pixel brightness is b .

The halftone image is then defined as:

$$[B(x,y)] = \begin{bmatrix} b(0,0) & b(0,1) & \dots & b(0,H) & -1 \\ b(1,0) & b(1,1) & \dots & b(1,H) & -1 \\ \vdots & \vdots & & \vdots & \\ b(V-1,0) & b(V-1,1) & \dots & b(V-1,H) & -1 \end{bmatrix} \quad (1)$$

In order to make the calculation of the pyramidal image decomposition easier, the matrix is divided into K blocks (sub-images) of size $m \times m$ ($m = 2^n$) and on each is then built a multi-layer IDP.

The number p of the IDP layers for every block is in the range $0 \leq p \leq n-1$. The case $p = n-1$ corresponds to complete pyramidal decomposition comprising maximum number of layers, for which the image is restored without errors (all decomposition components are used).

The IDP top (layer $p = 0$) for a block of size $2^n \times 2^n$ contains coefficients, from which after inverse transform is obtained its worse (coarse) approximation.

The next IDP layer for the same block (the layer $p = 1$) is defined from the difference between the block matrix and the approximation, divided into 4 sub-matrices of size $2^{n-1} \times 2^{n-1}$ in advance.

The highest IDP layer (layer $p = n-1$) is based on the information from the pixels in all the 4^{n-1} difference sub-matrices of size 2×2 , obtained in result of the $(n-1)$ -time division of the initial matrix into sub-matrices.

In correspondence with the described principle, the matrix $[B_{k_0}]$ of one image block could be represented as a decomposition of $(n + 1)$ components:

$$[B_{k_0}] = [\tilde{B}_{k_0}] + \sum_{p=1}^{n-1} [\tilde{E}_{k_{p-1}}] + [E_{k_n}]$$

for $k_p = 1, 2, \dots, 4^p K$ and $p = 0, 1, \dots, n-1$. (2)

Here k_p is the number of the sub-matrices of size $m_p \times m_p$ ($m_p = 2^{n-p}$) in the IDP layer p ; the matrices $[\tilde{B}_{k_0}]$ and $[\tilde{E}_{k_{p-1}}]$ are the corresponding approximations of $[B_{k_0}]$ and $[E_{k_{p-1}}]$; $[E_{k_n}]$ is the matrix, which represents the decomposition error in correspondence with Eq. (2), for the case, when only the first n components are used.

The matrix $[E_{k_{p-1}}]$ of the difference sub-block k_{p-1} in the IDP layer p is defined as:

$$[E_{k_{p-1}}] = [E_{k_{p-2}}] - [\tilde{E}_{k_{p-2}}], \text{ for } p = 2, 3, \dots, n-1. \text{ In this case } p = 1: \quad (3)$$

$$[E_{k_0}] = [B_{k_0}] - [\tilde{B}_{k_0}] \quad (4)$$

The matrix $[E_{k_{p-1}}]$ of the difference sub-block in the layer p is divided into $4^p K$ sub-matrices $[E_{k_p}]$ and for each is then calculated the corresponding approximating matrix $[\tilde{E}_{k_p}]$. The matrices $[\tilde{E}_{k_p}]$ for $k_p = 1, 2, \dots, 4^p K$ define the next decomposition component $(p + 1)$, represented by Eq. (2). For this is necessary to calculate the new difference matrix and then to perform the same operations again following the already presented order.

2.3. Image Representation with PNN

The new method for image representation is based on the IDP decomposition, in which the direct and inverse transforms in all layers are performed using 3-layer BPNN [13].

The general BPNN structure in PNN was chosen to be a 3-layer one of the kind $m^2 \times n \times m^2$, shown in Fig. 1. The input layer is of m^2 elements, which correspond to the input vector components; the hidden layer is of n elements for $n < m^2$, and the output layer is of m^2 elements, which correspond to the output vector components.

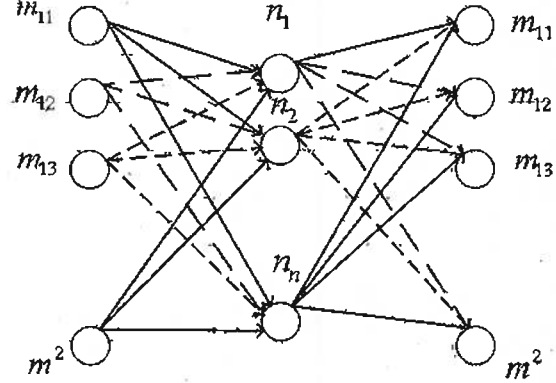


Figure 1: Three-layer BPNN of $n < m^2$ Neurons in the Hidden Layer and m^2 Neurons in the Input and in the Output Layer

The input m^2 -dimensional vector is obtained in result of the transformation of the elements m_{ij} of each image block of size $m \times m$ into one-dimensional massif of length m^2 following the "meander" scan, shown in Fig. 2.

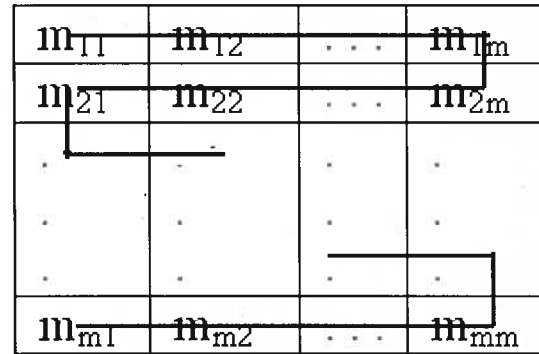


Figure 2: The "Meander" Scan Applied on Image Block of Size $m \times m$

In order to obtain better compression the processed image is represented by the sequence of m^2 -dimensional vectors $\vec{X}_1, \vec{X}_2, \dots, \vec{X}_K$, which are then transformed into the n -dimensional vectors $\vec{h}_1, \vec{h}_2, \dots, \vec{h}_K$ correspondingly.

The components of the vectors \vec{h}_k for $k = 1, 2, \dots, K$ represent the neurons in the hidden layer of the trained 3-layer BPNN with $m^2 \times n \times m^2$ structure. In the output NN layer the vector \vec{h}_k is transformed back into the m^2 -dimensional output vector \vec{Y}_k , which approximates the corresponding input vector \vec{X}_k .

The approximation error depends on the training algorithm and on the participating BPNN parameters. The training vectors $\bar{X}_1, \bar{X}_2, \dots, \bar{X}_K$ at the BPNN input for the PNN layer $p = 0$ correspond to the image blocks.

For the training was chosen the algorithm of Levenberg-Marquardt (LM) [13, 18], which ensures rapidity in cases, when high accuracy is not required and as a result is suitable for the presented approach.

One more reason is that the data necessary for the training has significant volume and information redundancy, but this does not make worse the training with the LM algorithm and influences only the time needed (i.e. it becomes longer).

The parameters of the 3-layer BPNN define the relations between the inputs and the neurons in the hidden layer, and between the neurons from the hidden and the output layer. These relations are described using weight matrices and vectors, which contain threshold coefficients, and with functions for non-linear vector transform.

The relation between the input m^2 -dimensional vector \bar{X}_k and the corresponding n -dimensional vector \bar{h}_k in the hidden BPNN layer for the PNN layer $p = 0$ is:

$$\bar{h}_k = f([W]_1 \bar{X}_k + \bar{b}_1) \text{ for } k=1,2,\dots,K, \quad (5)$$

where $[W]_1$ is the matrix of the weight coefficients of size $m^2 \times n$, which is used for the linear transform of the input vector \bar{X}_k ; \bar{b}_1 is the n -dimensional vector of the threshold coefficients in the hidden layer, and $f(x)$ is a linear activating sigmoid function, defined by the relation:

$$f(x) = 1/(1 + e^{-x}). \quad (6)$$

In result the network performance becomes partially non-linear and this influence is stronger when x is outside the range $[0, +1]$.

The relation between the n -dimensional vector \bar{h}_k of the hidden layer and the m^2 -dimensional BPNN vector \bar{Y}_k from the PNN layer $p = 0$, which approximates \bar{X}_k , is defined in accordance with Eq. (5) as follows:

$$\bar{Y}_k = f([W]_2 \bar{h}_k + \bar{b}_2) \text{ for } k=1,2,\dots,K, \quad (7)$$

where $[W]_2$ is a matrix of size $n \times m^2$ representing the weight coefficients used for the linear transform in the hidden layer of the vector \bar{h}_k , and \bar{b}_2 is the m^2 -dimensional vector of the threshold coefficients for the output layer.

Unlike the pixels in the halftone images, whose brightness is in the range $[0,255]$, the components of the

input and output BPNN vectors are normalized in the range $x_i(k), y_j(k) \in [0, 1]$ for $i = 1,2,\dots, m^2$.

The components of the vector which represents the neurons in the hidden layer $h_j(k) \in [0,1]$ for $j = 1,2 \dots, n$ are placed in the same range, because they are defined by the activating function $f(x) \in [0,1]$. The normalization is necessary, because it enhances the BPNN efficiency [18].

The image representation with PNN is performed in two consecutive stages:

- BPNN training;
- Coding of the obtained output data.

For the BPNN training in the PNN layer $p = 0$, the vectors \bar{X}_k are used as input and reference ones, with which are compared the corresponding output vectors. The comparison result is used to correct the weight and the threshold coefficients so that to obtain minimum MSE. The training is repeated until the Mean Square Error (MSE) value for the output vectors becomes lower than predefined threshold.

For the training of the 3-layer PNN in the next ($p > 0$) PNN layers are used the vectors obtained after the dividing of the difference block $[E_{k,p}]$ (or sub-block) into $4^p K$ sub-blocks and their transformation into corresponding vectors.

The BPNN training for each layer $p > 0$ is performed in the way already described for the layer $p = 0$.

In the second stage the vectors in the hidden BPNN layers for all PNN layers are coded losslessly with entropy coding.

The coding is based on two methods [3]:

- Run-Length Coding (RLC);
- Variable length Huffman coding.

The block diagram of the pyramid decomposition for one block of size $m \times m$ with 3-layer BPNN for the layers $p = 0,1,2$ and entropy coding/decoding is shown in Fig. 3.

When the BPNN training is finished, for each layer p are defined the corresponding output weight matrix $[W]_p$ and the threshold vector $[b]_p$.

The entropy coder (EC) compresses the data transferred to the decoder for the layer p , i.e.:

- The vector of the threshold coefficients for the neurons in the output NN layer (common for all blocks in the layer p);
- The matrix of the weight coefficients of the relations between the neurons in the hidden layer towards the output BPNN layer (common for all blocks in the layer p);

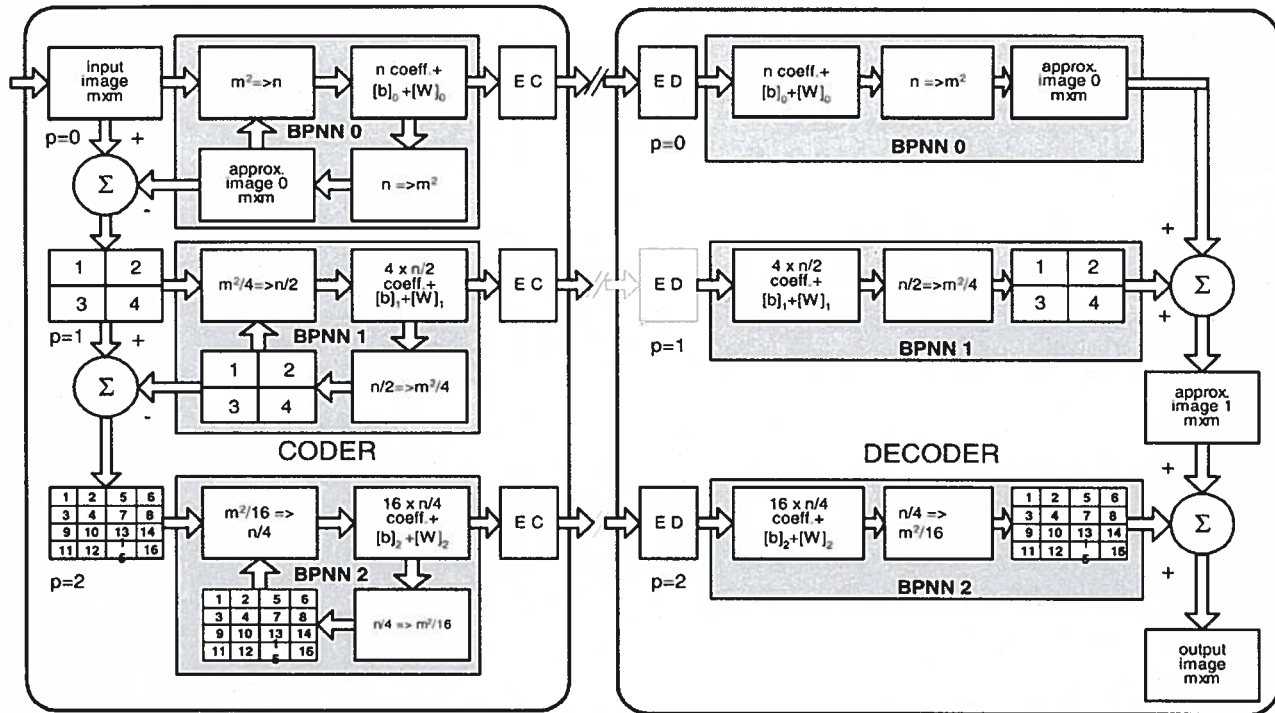


Figure 3: Block Diagram of the 3-layer Inverse Pyramidal Image Decomposition with 3-layer BPNN. Here $[b]_p$ – the Vector of the Threshold Coefficients in the Output Layer for $p = 0,1,2$; $[W]_p$ – the Matrix of the Weight Coefficients between the Hidden and the Output BPNN Layers for $p = 0,1,2$

- The vector of the neurons in the hidden BPNN layer, personal for each block in the layer p .

In the decoder is performed the entropy decoding (ED) of the compressed data. After that the BPNN in the layer p is initialized setting the values of the threshold coefficients for the neurons in the output layer and of the weight coefficients for the neurons, connecting the hidden and the output layers.

At the end of the decoding the vector of the neurons in the hidden BPNN layer for each block is transformed into corresponding output vector. The obtained output vectors are used for the restoration of the processed image.

3. EXPERIMENTAL RESULTS

The experiments were performed on the basis of the PNN simulation with MATLAB. For the simulation is necessary to perform the following preliminary operations:

- Transformation of the input data into a sequence of vectors;
- Selection of the PNN structure;
- PNN creation and initialization of its parameters;
- BPNN training using the input vectors so that to obtain the required output vectors;
- Testing of the PNN algorithm with various test images and evaluation of their quality after restoration (objective and subjective).

The PNN algorithm consists of two basic steps [15]: coding; and decoding.

An example test satellite image of size 8192x8192 pixels, used for the experiments is shown in Fig. 4.



Figure 4: Panchromatic Band of the Satellite Image of the High Resolution (Ville de Sherbrooke, Quebec, Canada - colored Composition with Visible Bands R.G.B. (4m) Merged in the Panchromatic Band (1m) Acquired on May 20th, 2001 at 10 h 50. Projection: UTM North America on 1983 (in meters)

The experiments with the PNN algorithm simulation were performed with 12 satellite test images of size 1024×1024 , 8 bpp (i.e. 1 048 576 B), which are parts of the image in Fig. 4. The used original test images are shown in Fig. 5.

In the PNN layer $p = 0$ the image is divided into K blocks of size 16×16 pixels, ($K = 4096$). At the BPNN input for the PNN layer $p = 0$ is passed the training matrix of the input vectors of size $256 \times 4096 = 1\,048\,576$ B. In

the hidden BPNN layer the size of each input vector is reduced from $m^2 = 256$ to $n = 16$.

The restoration of the output vector in the decoder is performed using the 16 components of the hidden layer in the neural network (matrix of vectors) together with the vector of the threshold values (a threshold matrix) and the matrix of the weight coefficients (a weight matrix) in the BPNN output layer (Table 1).

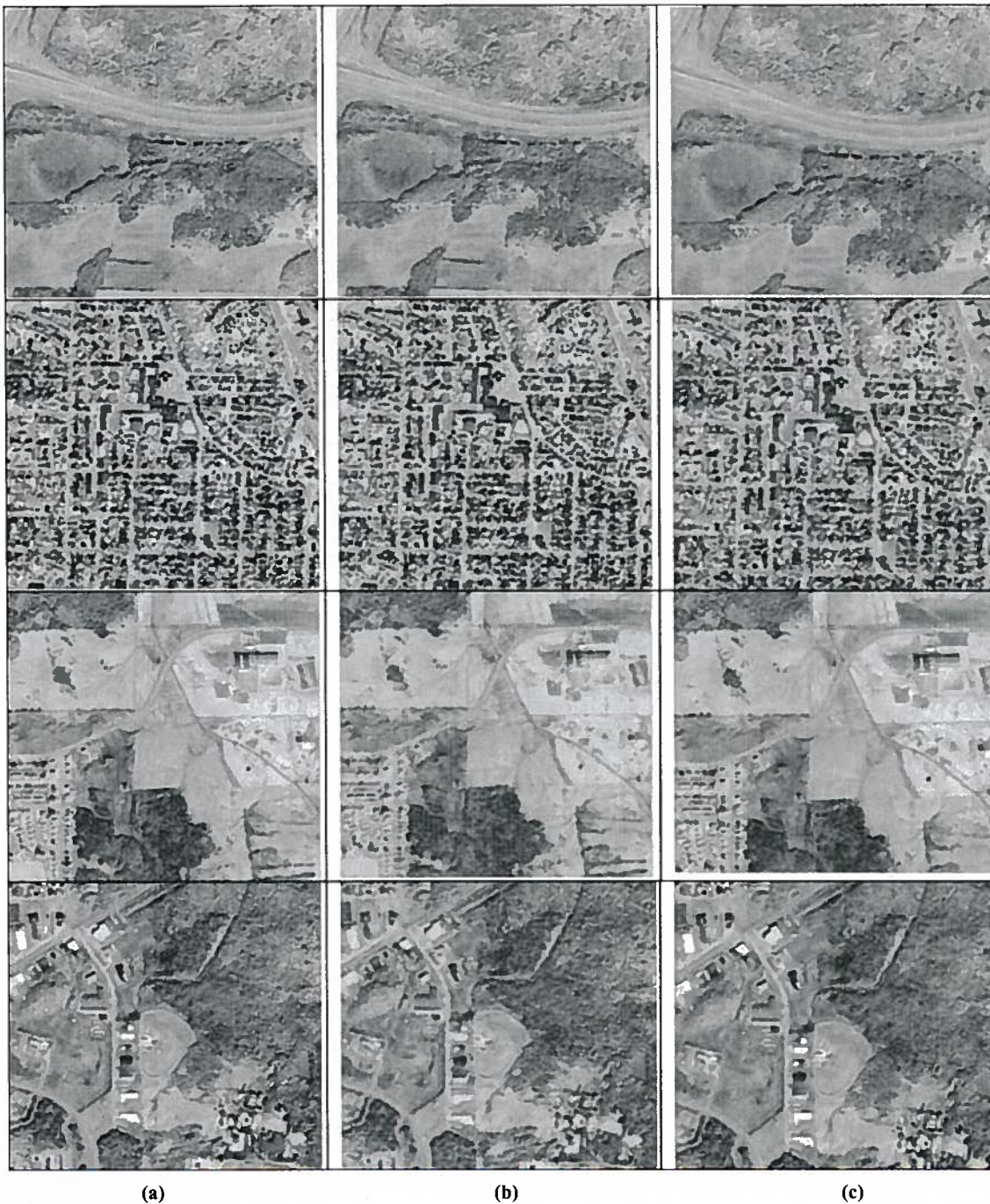


Figure 5: The Results Obtained for all 12 test Satellite Images of the High Resolution 1 048 576 Bytes for Each Image CR = 106.11 ÷ 108.29. (a) Original Images; (b) PNN Method; (c) LuraWave JPEG2000 Method

Multiple tests of compression for the presented structure of the neural network for the corresponding data show the average results of compression ratio (CR) for each of three matrix of representation: 64 : 15.25 : 22.5, respectively. As it was already pointed out, the data has high correlation and is efficiently compressed with entropy coding.

For the layer $p = 0$ the size of the transformed data obtained with neural network presentation for the satellite test images is 559 104 B (before the entropic coding), i.e. – two times less than that of the original image (1 048 576 B). For example, the compressed data size for the same layer ($p=0$) of the test image *Sherbrooke_Pan1.tif* is 9773.9 B (the result is given in Table 1).

Table 1
Structure of Data Presentation in the Space of the Neural Network Hidden Layer for the Test Image *Sherbrooke_Pan_1.tif*

<i>PNN, p = 0 image size</i> <i>1024 × 1024</i>	<i>Matrix of vectors</i> <i>4096 sub-images</i>	<i>Threshold Matrix</i>	<i>Weight matrix</i>	<i>File size [B]</i>
Structure size	16 × 4096 × 8	256 × 1 × 8	256 × 16 × 8	
Total bytes number	524 288	2048	32 768	559 104
CR for each structure	64	15,28	22,63	
Total bytes number after compression	8 192	133.99	1 448	9 773

Compression ratio (CR) = 107,28

Taking into account the size of the original image, is calculated the compression ratio $CR = 107.28$. The Peak Signal to Noise Ratio for the first test image *Sherbrooke_Pan1.tif* for $p=0$ (Table 1) is $PSNR = 21.38$ dB, and the $RMSE = 107.06$. In the same table are given the compression ratios obtained with PNN for additional 12 test images of same size (1024 × 1024). It is easy to see that for the mean compression ratio $CR = 107.32$ is obtained $PSNR > 22.24$ dB, i.e. the visual quality of the restored test images is suitable for various applications. Better image quality is obtained when the next pyramid layers are added [16, 17].

In Table 2 are given the results for the group of 4 satellite test images after PNN compression, implemented

with MATLAB. The results obtained (Table 3) show that the PNN method performance for the group of similar test images surpasses that for single images (Table 2). Thus, the results obtained show higher compression ($CR = 121.93$) for a group of images in comparison with the average compression ratio ($CR = 107.06$) obtained for each image separately.

The NN architecture used for the experiments for the zero level comprises 256 neurons in the input layer, 16 neurons in the hidden layer, and 256 neurons in the output layer. The chosen ratio for the input vectors was correspondingly: 80% for *Training*; 10 % for *Validation* and 10% for *Testing*.

Table 2
The Results Obtained for all 12 High-resolution Test Satellite Images (1 048 576 Bytes each) and the Comparison Result for *LuraWave* (JPEG2000 Compression)

<i>File Name</i>	<i>PNN</i>			<i>LuraWave (JPEG 2000)</i>		
	<i>CR</i>	<i>PSNR [dB]</i>	<i>Compressedfile [B]</i>	<i>CR</i>	<i>PSNR [dB]</i>	<i>Compressedfile [B]</i>
1	107,28	21,38	9 774	107,15	22,58	9 786
2	106,11	18,25	9 882	106,12	19,09	9 881
3	107,61	20,31	9 774	107,61	20,89	9 744
4	108,48	22,39	9 666	126,02	24,05	8 321
5	107,99	18,80	9 710	108,00	19,61	9 709
6	108,28	21,08	9 684	108,29	22,16	9 683
7	107,17	18,65	9 784	107,18	20,46	9 783
8	105,51	20,24	9 938	105,54	22,03	9 936
9	107,37	20,04	9 766	107,39	21,87	9 764
10	106,87	19,65	9 812	106,89	21,86	9 810
11	107,46	19,86	9 758	107,06	22,17	9 794
12	107,92	20,73	9 716	107,93	21,72	9 715

Table 3
Results Obtained for a Group of 4 Satellite Test Images after PNN Compression (4 194 304 B for this group; Fig. 6) and Comparative Data with LuraWave JPEG2000 and LuraWave Compression format for Different Level of Compression (CR): 121.93 : 159.15 : 273.6, Respectively

	<i>CR = 121,93</i>		<i>CR = 159,15</i>		<i>CR = 273,6</i>	
	<i>PSNR [dB]</i>	<i>File size [B]</i>	<i>PSNR [dB]</i>	<i>File size [B]</i>	<i>PSNR [dB]</i>	<i>File size [B]</i>
PNN	22,73	34 400	21,03	26 354	19,34	15 330
LuraWaveJPEG 2000 (.jpg)	21,79	34 398	21,17	26 354	20,04	15 328
LuraWaveformat (.lwf)	21,18	34 396	21,61	26 352	19,61	15 328

In the next experiment we compared a sequence of 4 satellite test images of size 2048x2048, 8 bpp (4 194 304 B). We used the same PNN structure and the same learning algorithm. The database contained 16384 examples of 16x16 matrices. The group of static images presented on Fig. 6 creates an image of size 1024 × 1024 × 4, that is 4 194 304 B.

For visual evaluation only, in Fig. 7 are shown enlarged parts (256 × 256 pixels) of the test images from Fig. 5, obtained after compression with CR = 127. For the comparison are given the results obtained for PNN and LuraWave format, *lwf* (the results for *lwf* and JPEG 2000 are very close and for better presentation the result for JPEG 2000 is not included).

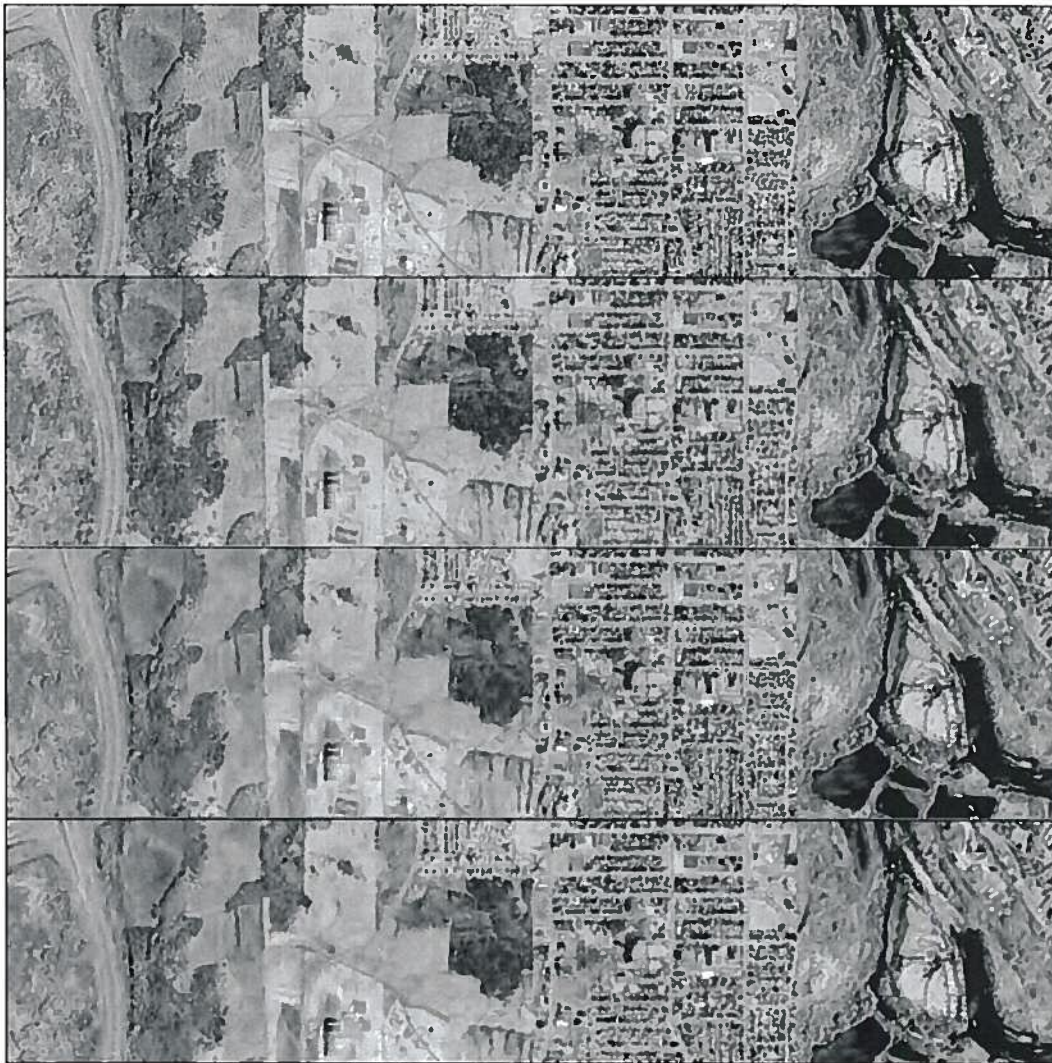


Figure 6: Results of Comparison with Three Methods, CR = 273.9: (a) Original Group of Similar Images; (b) Approximate Group of Similar Images by PNN; (c) Approximate Group of Similar Images by LuraWave SmartCompress 3.0 -JPEG2000; (d) LuraWave format (.lwf)

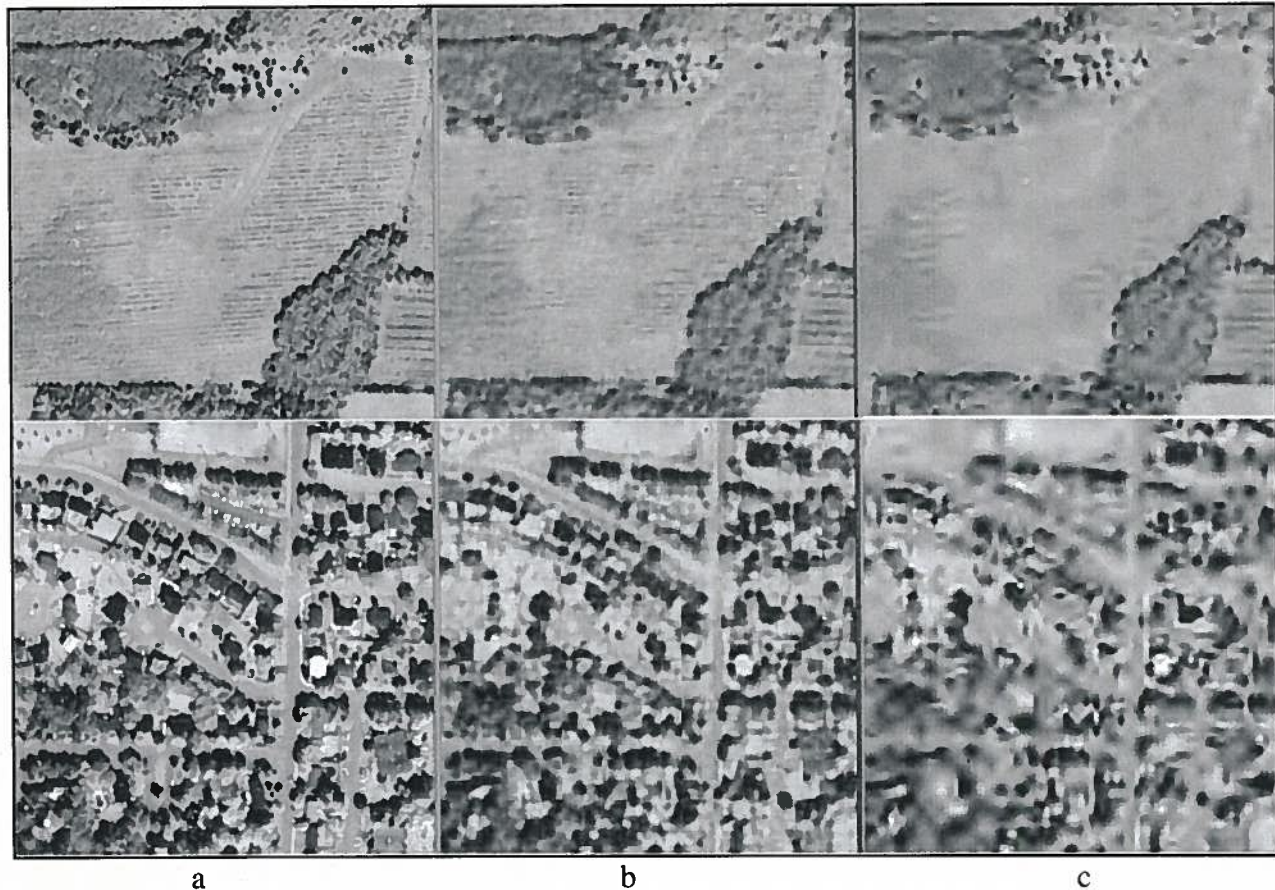


Figure 7: Enlarged Parts of the Test Images: (a) Original; (b) PNN Method; (c) LuraWave (*lwf*)

4. CONCLUSION

In this paper is presented one new approach for satellite image representation with pyramidal neural structure. The algorithm modeling was used to evaluate its performance for the group of 12 test images of high resolution. The results obtained show that for the tested group of images it ensures higher compression ratio (121.93) than for the case when they were processed individually (CR=107.06).

The PNN method is asymmetric (the coder is more complicated than the decoder) and this determines it mostly in application areas which do not require real time processing i.e. applications, for which the training time is not crucial. The computational complexity of the method depends on the training method selected.

The new method offers wide opportunities for application areas in the digital image processing, such as the progressive transfer via Internet, saving and searching in large image databases, remote sensing, etc.

The results obtained in this work confirm the theoretical presumption, that images of high resolution (satellite images, multi-spectral and hyper-spectral images) possess high degree of data redundancy. This allows the efficient

use of the PNN method for compression of high resolution images in remote sensing applications.

The results of the new approach implementation show sufficient visual quality of restored high-resolution satellite images. For higher compression ratios the visual difference between the restored images obtained with the PNN method and with other competitive compression methods are smaller.

Acknowledgment

This paper was supported by the CARTEL - Centre d'Application et de Recherche en Teledetection, Universite de Sherbrooke, Canada, <http://www.usherbrooke.ca/cartel/> and by the National Fund for Scientific Research of the Bulgarian Ministry of Education and Science (Contr. VU-I 305/2007)

References

- [1] A. Namphol, *et al.*, *Image Compression with a Hierarchical Neural Network*, IEEE Trans. on Aerospace and Electronic Systems, 32(1), 1996, 327-337.
- [2] A. Aboshosha (ed.), *International Journal on Graphics, Vision and Image Processing (GVIP), Special Issue on Image Compression*, www.icgst.com, 2007.
- [3] D. Salomon, *Data Compression*, Springer, 2004.

- [4] J. Jiang. Image Compressing with Neural Networks - A Survey, *Signal Processing: Image Communication*, Elsevier, **14**(9), 1999, 737-760.
- [5] K. Rao, P. Yip, Ed. *The Transform and Data Compression Handbook*. CRC Press LLC, 2001.
- [6] M. Barni, Ed., *Document and Image Compression*. CRC Press Taylor and Francis Group, 2006.
- [7] N. Kouda, *et al.*, Image Compression by Layered Quantum Neural Networks, *Neural Processing Letters*, **16**, 2002, 67-80.
- [8] R. Dony, S. Haykin, Neural Network Approaches to Image Compression, *Proceedings of the IEEE*, **23**(2), 1995, 289-303.
- [9] R. Gonzales, R. Woods, *Digital Image Processing*, Prentice Hall, 2002.
- [10] R. Gonzalez, R. Woods, S. Eddins, *Digital Image Processing using MATLAB*, Prentice Hall, 2004.
- [11] R. Kountchev, V. Haese-Coat, J. Ronsin. Inverse Pyramidal Decomposition with Multiple DCT, *Signal Processing. Image Communication*, Elsevier **17**, 2002, 201-218.
- [12] S. Kulkarni, B. Verma, M. Blumenstein, Image Compression Using a Direct Solution Method Based Neural Network, 10-th Australian Joint Conference on AI, Perth, Australia, 1997, 114-119.
- [13] St. Perry, H. Wong, L. Guan. *Adaptive Image Processing: A Computational Intelligence Perspective*, CRC Press LLC, 2002.
- [14] T. Acharya, P. Tsai, *JPEG2000 Standard for Image Compression*, John Wiley and Sons, 2005.
- [15] V. Cherkashyn, R. Kountchev, D.-C. He, R. Kountcheva. Adaptive Image Pyramidal Representation, *IEEE Symposium on Signal Processing and Information Technology (ISSPIT)*, 2008, 441-446.
- [16] V. Cherkashyn, N. Hikal, R. Kountchev, Y. Biletskiy. Image Compression based on the Invers Difference Pyramid with BPNN, *Proc. of IEEE International Symposium on Industrial Electronics (ISIE'06)*, 2006, 645-648.
- [17] V. Cherkashyn, R. Kountchev, Dong-Chen He. Image Decomposition on the basis of an Inverse Pyramid with 3-layer Neural Networks. *Proc. of IEEE International Joint Conferences on Computer, Information and Systems Sciences and Engineering (CISSE 2008)*. Bridgeport, USA, Springer. Dec. 2008.
- [18] Y. H. Hu, J.N. Hwang (Eds.). *Handbook of Neural Network Signal Processing*, CRC Press LLC, 2002.

**A Novel Adaptive Representation Method (AIPR/BPNN) of
Satellite Visible Images Very High Definition**

by

Valeriy Cherkashyn, Dong-Chen He
and Roumen Kountchev

Journal of Communication and Computer,

Vol. 7, No. 9 (Serial No. 70)

ISSN 1548-7709, USA, Sep. 2010, pp. 55-66

A Novel Adaptive Representation Method (AIPR/BPNN) of Satellite Visible Images Very High Definition

Valeriy Victorovich Cherkashyn¹, Dong-Chen He¹ and Roumen Kirilov Kountchev²

1. Centre for Research and Applications in Remote Sensing-CARTEL, University of Sherbrooke, Quebec J1K 2R1, Canada

2. Department of Radio Communication and Video Technologies, Technical University-Sofia, Sofia 1000, Bulgaria

Received: May 20, 2010 / Accepted: June 21, 2010 / Published: September 25, 2010.

Abstract: In this article the research of a new adaptive representation method on purpose to reduce the size of visible satellite images very high resolution is presented. The development of high resolution remote sensing systems demands new intelligent approaches of acquisition, transmission, and storage of the received data. Such approaches for enormous data volumes ask for the aid of data compression. We offer a novel adaptive technique of image representation. This method is based on an inverse pyramidal decomposition of original data and neural networks. The basic advantage of the pyramidal decomposition in comparison with the other methods for image compression is the ability to perform “progressive” transfer (or storage) for every consecutive decomposition layer. In result, the image could be restored with high compression ratio and gradually improving quality. The transformed image is presented by values of coefficients of the hidden layer of a neural network. They are calculated during the period of training of three layer back propagation neural network (BPNN). A novel original method of progressive transformation to panchromatic band of IKONOS images has been applied. The results of simulation of a new method showed the best visual quality of the restored images comparison with JPEG2000.

Key words: Image compression, adaptive inverse pyramidal representation (AIPR), pyramidal neural network (PNN).

1. Introduction

The data files of satellite images very high definition, multispectral and hyper spectral represent extremely large volumes. However, transmission media have a finite and limited bandwidth. It leads to serious problems for gathering, storage and data transmission.

For example, one working minute of Ikonos remote sensing sensor demands 2 Gb (gigabyte) of storage memory. To keep a single image of the Quebec territory, Canada it takes 3.4 Tb (terabyte). The

increased data volume inherent to the high resolution data of Ikonos is a significant overhead on transmission, processing and storage systems. Obviously, without any form of compression the amount of storage required for a modest size of digital library of images would be staggeringly high. To decrease this volume it's useful to use the data representation (compression) methods.

Data compression may be beneficial to both data storage and data transmission. Data compression for remote sensing images is effective, because, as natural images, they exhibit large spatial correlation. Therefore, new methods exploiting both the spatial and the spectral redundancy have recently appeared. The compression is achieved by the removal of one or more of three basic redundancies [1]:

(1) coding redundancy, which exists when less than optimal code words are used;

Corresponding author: Valeriy Victorovich Cherkashyn, Ph.D., research fields: image processing, intellectual systems, GIS, artificial intelligence, neural networks, data mining, remote sensing. E-mail: Valeriy.Cherkashyn@USherbrooke.ca.

Dong-Chen He, professor, Ph.D., research fields: image processing, classification, segmentation, mathematical morphology, GIS, remote sensing.

Roumen Kirilov Kountchev, professor, Ph.D., D.Sc., research fields: image processing, water marking, data mining, telecommunication.

(2) interpixel redundancy, which results from correlations between the pixels of an image;

(3) psychovisual redundancy, which is due to data that is ignored by the human visual system (i.e., visually nonessential information).

Data compression may be broadly divided in these three classes:

(1) lossless compression refers to the process whereby the decoded image after coding and decoding is exactly the same as the input image, and achieves compression factors of at most 4:1;

(2) near-lossless compression refers to the process whereby the decoded image after coding and decoding is perceptually equivalent to the input image, and achieves compression factors of about 30:1;

(3) lossy compression refers to the process whereby some information is discarded, leading to much higher compression ratios.

Usually, a lossy compression conveys high distortion rates at high compression rates.

1.1 Comparing the Different Methods of Data Representation

The classic methods [2-6] could be classified in the following main groups:

(1) deterministic and statistical orthogonal linear transforms (DFT, DCT, WHT (Walsh-Hadamard transform) and KLT (Karhunen-Loeve transform), SVD (Singular Value decomposition), PCA (Principal Component analysis) correspondingly);

(2) discrete wavelet transforms (DWT, EZW (Embedded Zerotree Wavelet), etc.);

(3) transforms based on prediction (LP, Adaptive LP, etc.);

(4) vector quantization (VQ, Adaptive VQ, Multistage Predictive VQ, etc.);

(5) fractal transforms (IFS, Quadtree Partitioned IFS, etc.);

(6) decompositions, based on various pyramids, such as GP/LP, RLP, RSP/RDP, IDP, etc.

The analysis of these methods shows that for image

compression are usually used deterministic orthogonal transforms and linear prediction with fixed coefficients [4, 6]. For example, the standard JPEG is based on discrete cosine transform (DCT) and linear prediction (LP), and the standard JPEG2000 [3] is based on the discrete wavelet transform (DWT). The transformed image data are compressed with entropy coding [6], implemented as a combination of various methods for lossless coding (RLC, AC (Arithmetic Coding), LZW (Lempel-Ziv-Welch encoding), etc.). The transform methods like DCT, DWT and LP are deterministic and the values of the matrices' coefficients do not depend on the content of the image. For this reason, the compression efficiency is low when the correlation between the image content and the corresponding transform functions is low. The statistical transforms [2] are more efficient than the deterministic ones, but they have higher computational complexity. The fractal methods has a similar disadvantage [3, 4], which together with this are not enough efficient when images with unclear texture regions are processed. The actual pyramidal image decompositions are usually implemented with digital filters with fixed decimation and interpolation coefficients or use some kind of transform, for example the Multiple DCT [5], i.e., these methods are not adapted well to the image content.

1.2 Computational Intelligence Techniques

An adaptive image processing systems use the class of algorithm commonly known as computational intelligence (CI) techniques [7-10] and has been used to refer to the general attempt to simulate human intelligence on computers, the so-called "artificial intelligence" (AI) approach [11]. The specific definition of CI includes neural network (NN) techniques [12-18]. A distinguishing characteristic of these algorithms is that they are either biologically inspired. A neural network consists of a large number of computational units or neurons which are massively interconnected with each other. Each such connection

between artificial neurons is characterized by an adjustable *weight* which can be modified through a training process such that the overall behaviour of the network is changed according to the nature of specific training examples provided.

A group of methods for image representation, based on the use of artificial neural networks (NN) [8, 13] had recently been developed. Unlike the classic methods, this approach is distinguished by higher compression ratios, because together with the coding, NN training is performed. The results already obtained show that these methods can not successfully compete the still image compression standards, JPEG and JPEG2000 [3]. For example, the Adaptive Vector Quantization (AVQ), is based on SOM NN (Self-organizing map neural network). This requires using the code books with too many vectors, needed to ensure high quality of the restored image and this results in lower compression. One of the first encouraging results for a similar method of research have been received in [19-21].

For the purpose optimization, we could embed the image model parameters as an adjustable network weights to be optimised through the network's dynamic action. The adaptive capability of neural networks through the adjustment of the network weights will prove useful in addressing the requirements of representation, segmentation, characterization and optimisation in adaptive image processing system design.

In this article a new adaptive method of inverse pyramidal decomposition of digital images with 3-layer BPNN's is offered. Results have received by the given method of modeling confirm a significant improvement of a visual quality of restored image in comparison with JPEG2000 method for all three pyramid levels.

2. Representation Method for Visible Satellite Images Very High Definition

2.1 Pyramidal Decomposition Selection

The basic advantage of the pyramidal decomposition in comparison with other methods for image compression is the ability to perform "progressive" transfer (or storage) of the approximating image obtained for every consecutive decomposition layer [19]. In result, the image could be first restored with high compression ratio and relatively low quality and permits quality improvement on request. The classic approach for progressive image transfer is based on the Laplasian pyramid (LP) combined with the Gaussian (GP).

In this paragraph is offered new approach for pyramidal image representation, based on the so-called *Adaptive Inverse Difference Pyramid (AIDP)* Fig. 1. Unlike the non-adaptive Inverse Difference Pyramid (IDP) it is built in the non-linear transformed image space using a group of NNs.

As with the transform coding techniques, AIDP

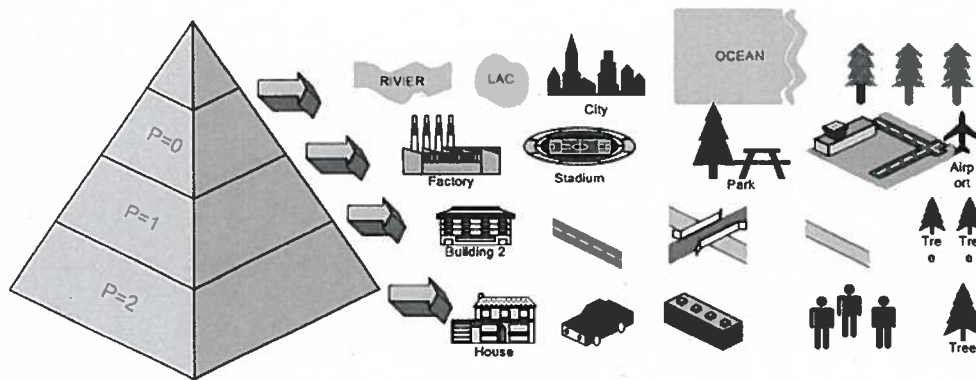


Fig. 1 Pyramidal representation of visible satellite images very high definition (VHD).

coding is based on the idea that the coefficients of a transform that correlates the pixels of an image can be coded more efficiently than the original pixels themselves. If the basis functions of the transform - in this case adaptive PNN (Pyramidal Neural Network) transformation - pack most of the important visual information into a small number of coefficients that in a hidden layer of AIDP are presented. The remaining coefficients can be quantized coarsely or truncated to zero with little image distortion. The AIDP is calculated starting the calculation from the pyramid top, placed down, and continues iteratively with the next pyramid layers.

The AIDP built this way, has some important advantages when compared to the LP: easier implementation of the progressive image transfer and compression enhancement in the space of the hidden layers of the corresponding NN.

2.2 Description of the Representation Method with an Inverse Difference Pyramid

The description of the given method has been presented in the articles [19, 21]. However we will offer method bases briefly. Thus, basically the description mathematically the digital image is usually represented as a matrix of size $H \times V$, whose elements $b(x, y)$ correspond to the image pixels; x and y define the pixel position as a matrix row and column and the pixel brightness is b . The halftone image is then defined as:

$$[B(x, y)] = \begin{bmatrix} b(0,0) & b(0,1) & \dots & b(0,H-1) \\ b(1,0) & b(1,1) & \dots & b(1,H-1) \\ \vdots & \vdots & & \vdots \\ b(V-1,0) & b(V-1,1) & \dots & b(V-1,H-1) \end{bmatrix} \quad (1)$$

In order to make the calculation of the pyramidal image decomposition easier, the matrix is divided into K blocks (sub-images) of size $m \times m$ ($m=2^n$) and on each is then built a multi-layer IDP. The number p of the IDP layers for every block is in the range $0 \leq p \leq n-1$. The case $p=n-1$ corresponds to complete pyramidal decomposition of maximum number of layers, for

which the image is restored without errors (all decomposition components are used).

The IDP top (layer $p=0$) for a block of size $2^n \times 2^n$ contains coefficients, from which after inverse transform is obtained its worse (coarse) approximation. The next IDP layer for the same block (the layer $p=1$) is defined from the difference between the block matrix and the approximation, divided into 4 sub-matrices of size $2^{n-1} \times 2^{n-1}$ in advance. The highest IDP layer (layer $p=n-1$) is based on the information from the pixels in all the 4^{n-1} difference sub-matrices of size 2×2 , obtained in result of the $(n-1)$ -time division of the initial matrix into sub-matrices.

In correspondence with the described principle, the matrix $[B_{k_0}]$ of one image block could be represented as a decomposition of $(n+1)$ components:

$$[B_{k_0}] = [\tilde{B}_{k_0}] + \sum_{p=1}^{n-1} [\tilde{E}_{k_{p-1}}] + [E_{k_n}] \quad (2)$$

for $k_p=1, 2, \dots, 4^p K$ and $p=0, 1, \dots, n-1$.

Here k_p is the number of the sub-matrices of size $m_p \times m_p$ ($m_p=2^{n-p}$) in the IDP layer p ; the matrices $[\tilde{B}_{k_0}]$ and $[\tilde{E}_{k_{p-1}}]$ are the corresponding approximations of $[B_{k_0}]$ and $[E_{k_{p-1}}]$; $[E_{k_n}]$ is the matrix, which represents the decomposition error in correspondence with Eq. (2), for the case, when only the first n components are used.

The matrix $[E_{k_{p-1}}]$ of the difference sub-block k_{p-1} in the IDP layer p is defined as:

$$[E_{k_{p-1}}] = [E_{k_{p-2}}] - [\tilde{E}_{k_{p-2}}] \quad (3)$$

for $p=2, 3, \dots, n-1$. In this case $p=1$:

$$[E_{k_0}] = [B_{k_0}] - [\tilde{B}_{k_0}] \quad (4)$$

The matrix $[E_{k_{p-1}}]$ of the difference sub-block in the layer p is divided into $4^p K$ sub-matrices $[E_{k_p}]$ and for each is then calculated the corresponding approximating matrix $[\tilde{E}_{k_p}]$. The submatrices $[\tilde{E}_{k_p}]$ for $k_p=1, 2, \dots, 4^p K$ define the next decomposition component $(p+1)$, represented by Eq. (2). For this is necessary to calculate the new difference matrix and then to perform the same operations again following the already presented order.

2.3 Data Image Representation with AIDP-BPNN

The new method for image representation is based on the IDP decomposition, in which the direct and inverse transforms in all layers are performed using 3-layer neural networks (Fig. 2) with error back propagation (BPNN) [21].

The general BPNN structure in AIDP was chosen to be a 3-layer one of the kind $m^2 \times n \times m^2$, shown in Fig. 2. The enter layer is of m^2 elements, which correspond to the input vector components; the hidden layer is of n elements for $n < m^2$, and the output layer is of m^2 elements as well, which correspond to the output vector components. The input m^2 -dimensional vector is obtained in result of the transformation of the elements m_{ij} of each image block of size $m \times m$ into one-dimensional massif of length m^2 using the "meander" scan.

In order to obtain better compression the processed image is represented by the sequence of m^2 -dimensional vectors $\vec{X}_1, \vec{X}_2, \dots, \vec{X}_K$, which are then transformed in the n -dimensional vectors $\vec{h}_1, \vec{h}_2, \dots, \vec{h}_K$ correspondingly. The components of the vectors \vec{h}_k for $k=1, 2, \dots, K$ represent the neurons in the hidden layer of the trained 3-layer BPNN with $m^2 \times n \times m^2$ structure. In the output NN layer the vector \vec{h}_k is transformed back into the m^2 -dimensional output vector \vec{Y}_k , which approximates the corresponding input vector \vec{X}_k . The approximation error depends on the training algorithm and on the participating BPNN parameters. The training vectors $\vec{X}_1, \vec{X}_2, \dots, \vec{X}_K$ at the BPNN input for

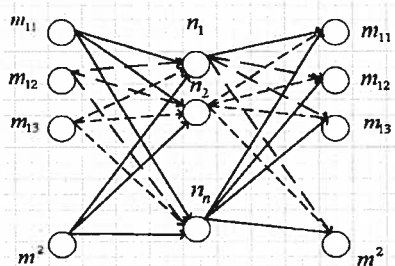


Fig. 2 Three-layer BPNN with $n_n < m^2$ neurons in the hidden layer and m^2 neurons in the input and in the output layer.

the AIDP layer $p=0$ correspond to the image blocks. For the training was chosen the algorithm of Levenberg-Marquardt (LM) [17, 18], which ensures rapidity in cases, when high accuracy is not required and as a result is suitable for the presented approach. One more reason is that the data necessary for the training has significant volume and information redundancy, but this does not make worse the training with the LM algorithm and influences only the time needed (i.e., it becomes longer).

The parameters of the 3-layer BPNN define the relations between the inputs and the neurons in the hidden layer, and between the neurons from the hidden and the output layer. These relations are described using weight matrices and vectors, which contain threshold coefficients, and with functions for non-linear vector transform.

The relation between the input m^2 -dimensional vector \vec{X}_k and the corresponding n -dimensional vector \vec{h}_k in the hidden BPNN layer for the AIDP layer $p=0$ is:

$$\vec{h}_k = f([W]_1 \vec{X}_k + \vec{b}_1) \text{ for } k=1, 2, \dots, K, \quad (5)$$

where $[W]_1$ is the matrix of the weight coefficients of size $m^2 \times n$, which is used for the linear transform of the input vector \vec{X}_k ; \vec{b}_1 is the n -dimensional vector of the threshold coefficients in the hidden layer, and $f(x)$ is a linear activating sigmoid function, defined by the relation:

$$f(x) = 1 / (1 + e^{-x}) \quad (6)$$

In result the network performance becomes partially non-linear and this influence is stronger when x is outside the range $[0, +1]$.

The relation between the n -dimensional vector \vec{h}_k of the hidden layer and the m^2 -dimensional BPNN vector \vec{Y}_k from the AIDP layer $p=0$, which approximates \vec{X}_k , is defined in accordance with Eq. (5) as follows:

$$\vec{Y}_k = f([W]_2 \vec{h}_k + \vec{b}_2) \text{ , for } k=1, 2, \dots, K, \quad (7)$$

where $[W]_2$ is a matrix of size $n \times m^2$ representing the weight coefficients used for the linear transform in the hidden layer of the vector \vec{h}_k , and \vec{b}_2 is the

m^2 -dimensional vector of the threshold coefficients for the output layer. Unlike the pixels in the halftone images, whose brightness is in the range $[0,255]$, the components of the input and output BPNN vectors are normalized in the range $x_i(k), y_i(k) \in [0,1]$ for $i=1,2,\dots,m^2$.

The components of the vector which represents the neurons in the hidden layer $h_j(k) \in [0,1]$ for $j=1,2,\dots,n$ are placed in the same range, because they are defined by the activating function $f(x) \in [0,1]$.

The normalization is necessary, because it enhances the BPNN efficiency [22].

The adaptive image representation with AIDP-BPNN method is performed in two consecutive stages:

- (1) BPNN training;
- (2) coding of the obtained output data.

For the BPNN training in the AIDP layer $p=0$, the vectors \bar{x}_k are used as input and reference ones, with which are compared the corresponding output vectors. The comparison result is used to correct the weight and the threshold coefficients so that to obtain minimum MSE.

The training is repeated until the MSE value for the output vectors becomes lower than predefined threshold.

For the training of the 3-layer BPNN in the next ($p>0$) AIDP layers are used the vectors obtained after the dividing of the difference block $[E_{k_{p-1}}]$ (or sub-block) into $4^p K$ sub-blocks and their transformation into corresponding vectors. The BPNN training for each layer $p>0$ is performed in the way already described for the layer $p=0$.

In the second stage the vectors in the hidden BPNN layers for all AIDP layers are coded losslessly with entropy coding. The coding is based on two methods: Run-Length Coding (RLC) and variable length Huffman coding. When the BPNN training is finished, for each layer p are defined the corresponding output weight matrix $[W]_p$ and the threshold vector $[b]_p$.

The entropy coder (EC) compresses the data transferred to the decoder for the layer p , i.e.:

The vector of the threshold coefficients for the neurons in the output NN layer (common for all blocks in the layer p);

The matrix of the weight coefficients of the relations between the neurons in the hidden layer towards the output BPNN layer (common for all blocks in the layer p);

The vector of the neurons in the hidden BPNN layer is personal for each block in the layer p . In the decoder is performed the entropy decoding (ED) of the compressed data. After that the BPNN in the layer p is initialized setting the values of the threshold coefficients for the neurons in the output layer and of the weight coefficients for the neurons, connecting the hidden and the output layers.

At the end of the decoding the vector of the neurons in the hidden BPNN layer for each block is transformed into corresponding output vector. The obtained output vectors are used for the restoration of the processed image.

The block diagram of the pyramid decomposition for one block of size $m \times m$ with 3-layer BPNN for the layers $p=0,1,2$ and entropy coding/decoding is shown in Fig. 3.

3. Simulation of the AIDP-BPNN Algorithm for Satellite Visible Images Very High Definition

For the simulation of the AIPR-BPNN algorithm is necessary to perform the following operations:

- transformation of the input data into a sequence of vectors;
- selection of the BPNN structure;
- BPNN creation and initialization of its parameters;
- BPNN training using the input vectors so that to obtain the required output vectors;
- testing of the AIDP-BPNN algorithm with various test images and evaluation of their quality after restoration (objective and subjective).

The AIDP-BPNN algorithm consists of the following four basic steps (for more details [21, 22]):

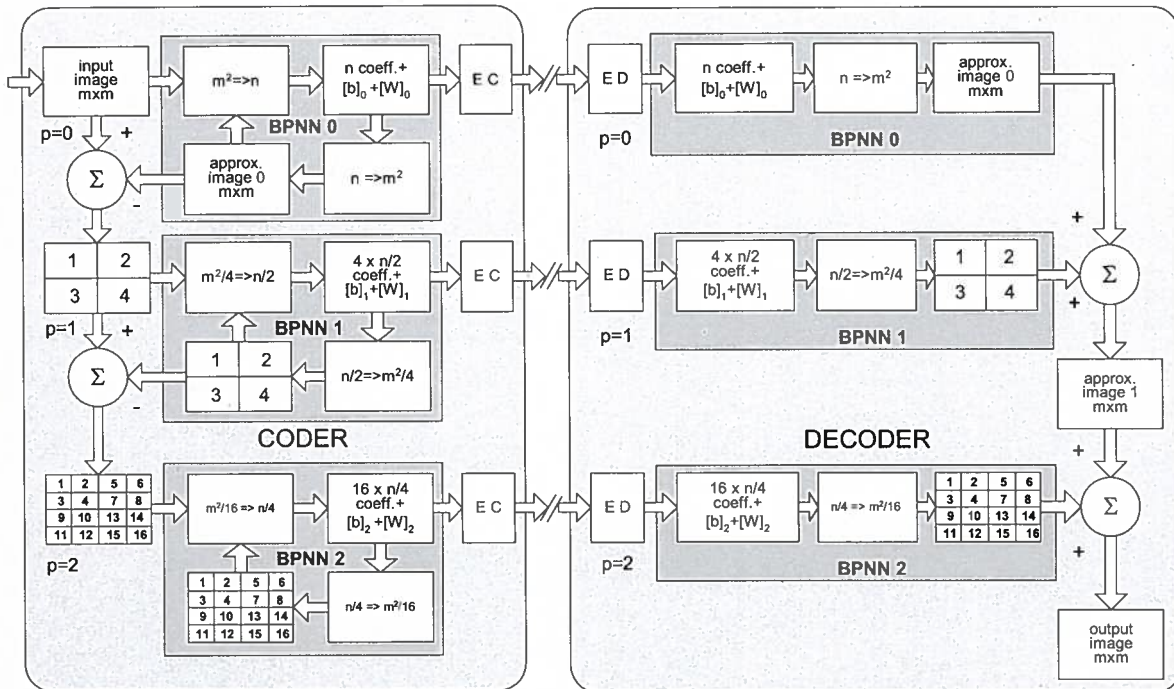


Fig. 3 Block diagram of the 3-layer inverse pyramidal image decomposition with 3-layer BPNN.

Here $[b]_p$ – the Vector of the Threshold Coefficients in the Output Layer for $p=0,1,2$; $[W]_p$ – the Matrix of the Weight Coefficients Between the Hidden and the Output BPNN Layer for $p=0,1,2$.

- (1) preliminary preparation of entrance data;
- (2) training of neural network;
- (3) coding;
- (4) decoding.

4. Experimental Results for Adaptive Inverse Pyramidal Neural Network

The experiments with the AIDP-BPNN algorithm were performed with still test satellite image (Fig.4.) (Sherbrooke, Quebec, Canada, May 20th, 2001 at 10h50) of size 1024×1024 , 8 *bpp* (i.e., 1 048 576B), and original image is presented in Fig. 4. On the first layer $p=0$ of the adaptive difference pyramids the image is divided into K blocks of size 16×16 pixels, ($K=4096$). For the layer $p=0$ of BPNN the input data consist of the training matrix of size 256×4096 , (1 048 576 B). The structure of a neural network that we chose: 256 neurons for enter level, 16 neurons for hidden level and 256 neurons for output level (Table 1).

The restored image in the decoder had performed

using matrix of 16 values of hidden layer ($16 \times 4096 \times 8$), together with the vector of the threshold values for output level of neural network (256×8) as well as the matrix of the weight coefficients in the output layer ($16 \times 256 \times 8$). For the layer $p=0$ the size of the data obtained is 559 104 B, i.e., half less than the original image (1 048 576B), approximately.

The effectiveness of a compression method with differential pyramidal data representation comes due to the fact that the entropy of error images (differential image in our case) is less than the original image. Besides, the data of these differential images are concentrated around 0 (see histogram, Fig. 5). As fact, for this type of data, it is better to use the entropic converter with variable length of code-words. The most important difference between traditional methods and AIPR consists in a calculation order (i.e., adaptive pyramidal calculation) of the image to be compressed. For the first step we calculate the image with high compression ratio and the low quality of image. Second



Fig. 4 (a) Original satellite image (1024x1024, 8bpp), Sherbrooke, Quebec, Canada, May 20th, 2001 at 10h50; (b) augmented fragment (red square) of the original satellite image.

Table 1 Compression Ratio (CR) dependence from the size of the original image and neural network structure.

The size of the image, [pixel]	The size of the subimage, [pixel]	Neural network structure (neuron quantity in a layer): enter : hidden : output	Compression ratio(CR)
256 × 256	8 × 8	68 : 8 : 64	59.43
512 × 512	16 × 16	256 : 16 : 256	61.44
1024 × 1024	16 × 16	256 : 16 : 256	100.72
2048 × 2048	32 × 32	1024 : 32 : 1024	125.39
4096 × 4096	32 × 32	1024 : 32 : 1024	203.11
8192 × 8192	64 × 64	4096 : 64 : 4096	253.36

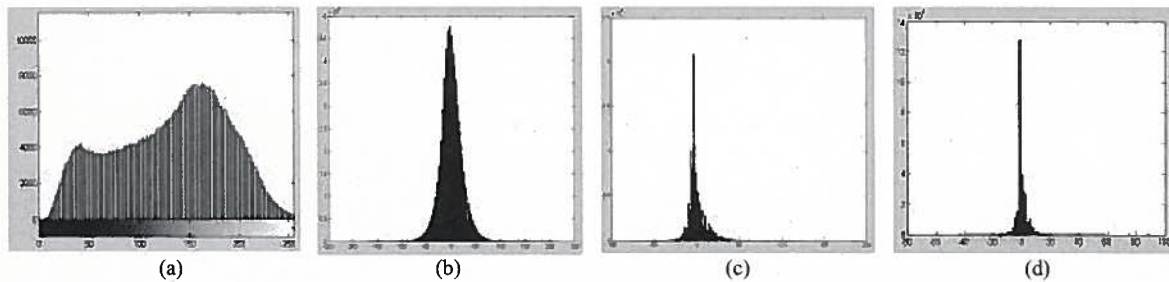


Fig. 5 Histograms for a) original image; b) differential image P=0; c) differential image P=1; d) differential image P=2.

step, we calculate the differential image for the subsequent level (P=01) of the pyramid. Next step; in the decoder's side (Fig. 3, on the right), we add restored differential image to ameliorate the quality of the restored image for level P=01. We repeat this kind of calculation for the next levels of the pyramid. Number of levels is chosen based on the condition of the problem, in practice, most empirically.

In the case of progressive transform the results of the

regression analysis (Fig. 6) shows the effectiveness of use of differential method of calculation for each levels of the pyramid. Thus, for the second level of pyramid (P=1) we see a big concentration of data around zero. In this case the entropic encoding gives favorable results for the compression ratio. This fact is essential to use the neural networks because the time of training diminishes for this type of data.

The regression results shown in Fig. 7 confirm the

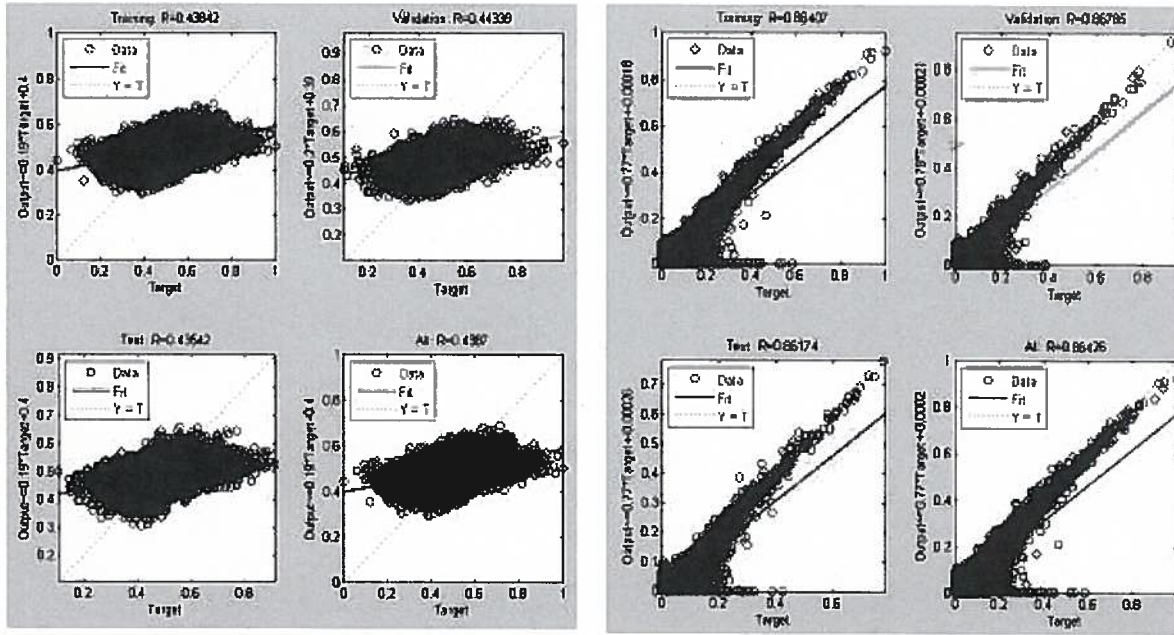


Fig. 6 Regression results for the pyramid levels P=0 and P=1.

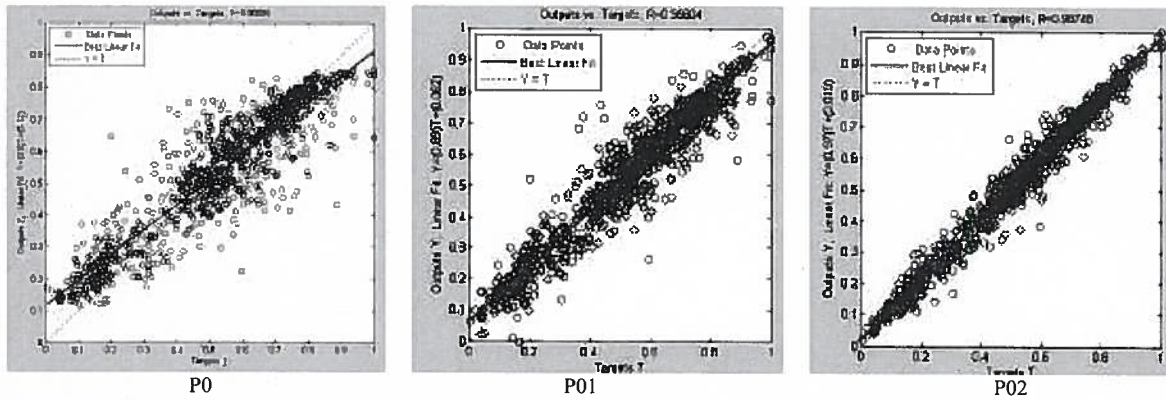


Fig. 7 Regression results for restored image for three levels of the adaptive pyramid neural networks.

progressive improvement of the image quality of the restored images for all of three levels of the pyramid.

We use the abbreviations for the following parameters of regression analysis (Table 2):

- M - Slope of the linear regression;
- B - Y intercept of the linear regression;
- R - Regression value ($R = 1$ means perfect correlation). For the restored image of the third level of the pyramid (P012=P0+P1+P2) the coefficient of retrogression is $R = 0.9875$.

Speaking about the latest applications of remote

Table 2 Evaluation results of the different parameters for three levels (P0; P01 and P012) of AIDP method.

Parameters	P=0	P=01	P=012
M	0.7991	0.8925	0.9670
B	0.1162	0.0620	0.0195
R	0.9090	0.9560	0.9875

sensing and their evaluation, the compression ratio (CR) parameter might not be the single parameter to evaluate the coding performance of the coding system. It has been shown that both the variance of the input data, nor the Signal-to-Noise Ratio (SNR), and not even the

Table 3 Evaluation results of the different parameters for three levels (P0; P01 and P012) of AIPR method.

Parameter	P=0, CR=106.26		P=01, CR=60.84		P=012, CR=40	
	AIPR	JPEG2000	AIPR	JPEG2000	AIPR	JPEG2000
PSNR, [dB]	20.89	19.09	24.45	20.75	27.48	22.26
RMSE	23.08	28.31	15.27	23.38	10.78	19.65
SSIM	0.9591	0.999	0.9917	1.0000	0.9948	1.0000

common Mean Squared Error (MSE) [23] inadequate thus other distortion measures have been proposed to help the evaluation.

The Structural Similarity (SSIM) index is a novel method for measuring the similarity between two images [24, 25]. The SSIM index can be viewed as a quality measure of one of the images being compared provided the other image is regarded as of perfect quality. The main ideas of SSIM were introduced in Ref. [25]. Suppose that x and y are local image patches taken from the same location of two images that are being compared.

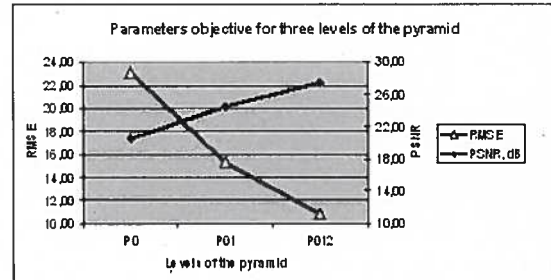
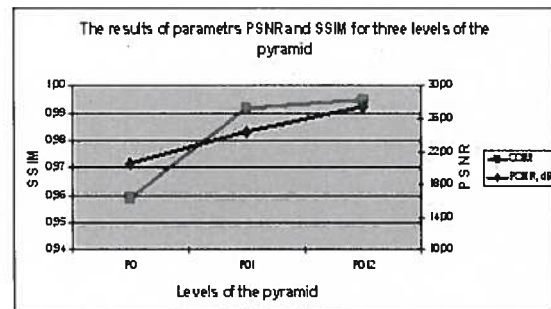
The local SSIM index measures the similarities of three elements of the image patches: the similarity $l(x, y)$ of the local patch luminances (brightness values), the similarity $c(x, y)$ of the local patch contrasts, and the similarity $s(x, y)$ of the local patch structures. These local similarities are expressed using simple, easily computed statistics, and combined together to form local SSIM.

The evaluation results of these three parameters in the Table 3 are presented.

It is necessary to underline the importance of the fact that for each level of the pyramid we have an increasing of the PSNR for 4 dB, from 20.89 dB then 24.45 dB to 27.48 dB (Fig. 8, Table 3). The RMSE of restored image have decreased by 13 points, from 23.08 for level, $P=0$ to 10.78 for level 3, $P=012$ (Table 3). The change of these parameters is essential and gives the hope of using this method in the future for the images of high definition.

5. Conclusions

In this paper a novel approach of the progressive transform coding of the visible satellites images very

**Fig. 8** Parameters PSNR and RMSE for three pyramid layers.**Fig. 9** Parameters PSNR and SSIM for three pyramid layers.

high definition is presented.

The results were obtained for the AIPR-BPNN algorithm. The original image was divided by 4096 sub-blocks of size 16×16 pixels for the first level ($p=0$). The second pyramid level was presented by the difference image (original image minus reconstructed image after the first level of transform) that was divided by 16384 sub-blocks of size 8×8 pixels. The third level was presented by the difference image error (difference image for the second level minus the reconstructed difference image error for the second level) that was divided by 65536 sub-blocks of 8 by 8 pixels.

As a result, we have received extremely good visual quality of restored satellite image compared with

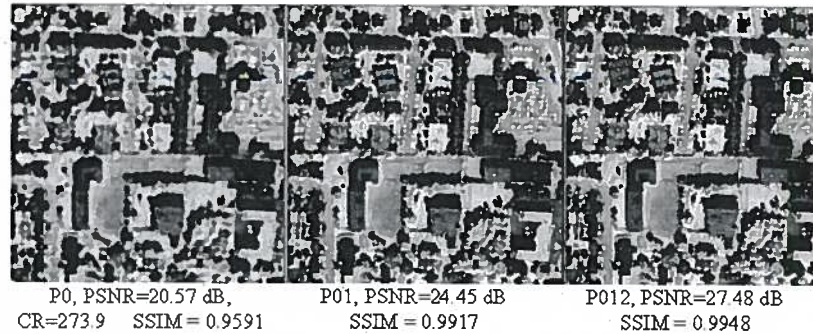


Fig. 10 Enlarged fragment for the three levels of restored image (P0, P01=P0+P1 and P012=P01+P2).

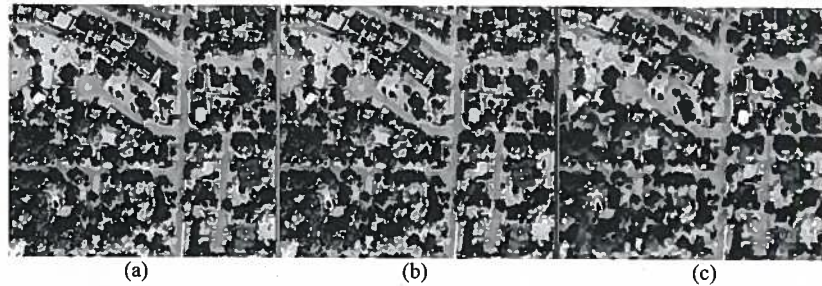


Fig. 11 Enlarged visual fragment for restored image (Table 3): a) original image; b) third pyramid level, AIPR-BPNN method; c) JPEG2000 method.

JPEG2000 at the same level of compression ratio (Fig. 10, 11, Table 3).

We have obtained very interesting results of SSIM parameter for both JPEG2000 and AIPR method (Fig. 9, Table 3). Despite the worst visual quality of restored images for JPEG2000 method, the SSIM parameter had a higher value than for AIPR method. It shows that SSIM parameter was adapted for the calculation of the local luminance changes, local contrasts changes, and local structures changes that presented for such kind of transformation like DWT, DCT and FFT. In our method we used the nonlinear algorithm of calculation which was based on artificial neural network. Thus, one of the reasons might be a nonlinear calculation based on neural network. So we have shown some restriction to use SSIM parameter.

We have received corroboration of the idea on that data presentation by AIPR-BPNN method essentially improves visual quality of the restored images. This improvement is more effective for satellite images very high definition.

The method AIPR-BPNN is an asymmetric method (the coder is more complicated than decoder). This determines it mostly in application areas which do not require real time processing i.e., applications, for which the training time isn't crucial.

But we can say with the hope that new methods of distributed computing, parallel calculation methods can reduce training time, and we could use this method extensively in the near future.

The hardware implementation of the method is beyond the scope of this work. The computational complexity of the method depends on the training method selected.

The new method offers wide opportunities for application in the field of digital image processing (multispectral and hyperspectral images; high definition and very high definition images); progressive transfer over the Internet; saving and searching in large image database; remote sciences; simulation for disaster management and with more efficient in parallel and distributed computing, etc.

Acknowledgments

This paper was supported by the CARTEL – Centre for Research and Applications in Remote Sensing, University of Sherbrooke, Quebec, Canada, by NSERC (Natural Sciences and Engineering Research Council of Canada) to D.C. He, and by the National Fund for Scientific Research of the Bulgarian Ministry of Education and Science (Contr. VU-I 305).

References

- [1] R. Gonzales, R. Woods, Digital Image Processing, 3rd ed., Prentice Hall, 2008.
- [2] M. Barni, Ed., Document and Image Compression. CRC Press Taylor and Francis Group, 2006.
- [3] T. Acharya, P. Tsai, JPEG2000 Standard for Image Compression, John Wiley and Sons, 2005.
- [4] K. Rao, P. Yip, Ed., The Transform and Data Compression Handbook, CRC Press LLC, 2001.
- [5] R. Kountchev, V. Haese-Coat, J. Ronsin, Inverse pyramidal decomposition with multiple DCT, Signal Processing : Image Communication, Elsevier 17(2002)201-218.
- [6] D. Salomon, Data Compression, Springer, 2004.
- [7] St. Perry, H. Wong, L. Guan, Adaptive Image Processing: a Computational Intelligence Perspective, CRC Press LLC, 2002.
- [8] M. Grana, R.J. Diro, Computational Intelligence for Remote Sensing, Berlin, Springer Press, 2008.
- [9] N. Ansari, E. Hou, Computational Intelligence for Optimization, Boston: Klumer Academic, 1997.
- [10] W. Pedrycz, Computational Intelligence: an Introduction, Boca Raton, FL : CRC Press, 1997.
- [11] P.H. Winston, Artificial Intelligence, Reading, M.A.: Addison-Wesley, 1984.
- [12] Y.H. Hu, J.N. Hwang (Eds.), Handbook of Neural Network Signal Processing, CRC Press LLC, 2002.
- [13] J. Jiang, Image compressing with neural networks - a survey, Signal Processing: Image Communication, Elsevier (1999)737-760.
- [14] N. Kouda, et al., Image compression by layered quantum neural networks, Neural Processing Lett. 16 (2002) 67-80.
- [15] Y.H. Hu, J.N. Hwang (Eds.), Handbook of Neural Network Signal Processing, CRC Press LLC, 2002.
- [16] R. Dony, S. Haykin, Neural network approaches to image compression, Proceedings of the IEEE 23 (1995) 289-303.
- [17] A. Namphol, et al., Image compression with a hierarchical neural network, IEEE Trans. on Aerospace and Electronic Systems 32 (1996) 327-337.
- [18] S. Kulkarni, B. Verma, M. Blumenstein, Image Compression using a direct solution method based neural network, 10th Australian Joint Conference on AI, Perth, Australia, 1997, pp. 114-119.
- [19] V. Cherkashyn, R. Kountchev, D.-C. He, R. Kountcheva, Adaptive image pyramidal representation, IEEE Symposium on Signal Processing and Information Technology, 2008.
- [20] V. Cherkashyn, D.-C. He, R. Kountchev, Compression of high-resolution satellite images with pyramidal neural network, International Journal of Neural Networks and Applications 2(2009) 5-14.
- [21] V. Cherkashyn, D.-C. He, R. Kountchev, Image decomposition basis of an inverse pyramid with 3-layer neural networks (forest images), Journal of Communication and Computer 6 (2009) 21-29.
- [22] V. Cherkashyn, N. Hikal, R. Kountchev, Y. Biletskiy, Image compression based on the inverse difference pyramid with BPNN, in: Proc. of IEEE International Symposium on Industrial Electronics (ISIE'06), 2006, pp. 645-648.
- [23] Z. Wang, A.C. Bovik, Mean squared error: love it or leave it? - A new look at signal fidelity measures, IEEE Signal Processing Magazine 26 (2009) 98-117.
- [24] Z. Wang, A. C. Bovik, H. R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Transactions on Image Processing 13 (2004) 600-612.
- [25] Z. Wang, A.C. Bovik, A universal image quality index, IEEE Signal Processing Lett 9 (2002) 81-84.

5. Discussion

Il n'existe encore aucun standard industriel régissant les techniques de compression par réseaux de neurones. Ce travail présente les divers essais de l'utilisation des réseaux de neurones vers la compression des images satellitaires de THRS. Les résultats prouvent la qualité visuelle concurrentielle des images compressées par notre méthode, en comparaison des méthodes courantes pour le niveau de la compression de 100 fois. De plus, les résultats encourageants étaient reçus pour le CR à 274 fois. Cela montre la capacité de cette méthode de s'adapter à l'augmentation continue du volume d'images de télédétection.

5.1. Formats actuels du marché géospatial

Dans le domaine de la télédétection, deux formats commerciaux sont davantage exploités actuellement : le format de compression MrSID de la firme LizardTech et ECW de la compagnie ERMapper. Ils utilisent une approche de compression par ondelettes et sont largement utilisés grâce au soutien de leurs sociétés. Nous avons démontré la faiblesse de la transformation d'ondelettes au CR de 274 fois pour des détails fins de l'image compressée.

5.2. Taille du bloc du traitement en fonction de la taille de l'image traitée

La taille du bloc de traitement a été fixée à 8 x 8 pixels (JPEG, JPEG2000). Cela a bien fonctionné pour des images de taille simple (jusqu'à 512 x 512). Malheureusement, l'augmentation de la grandeur du bloc de traitement (16 x 16 ou 32 x 32) pour des images simples amenait une altération de la qualité des images restaurées. Dans l'ère des images satellitaires où la taille des images est plus élevée (11 000 x 11 000 pixels), les vieilles méthodes exigent des dépenses calculatoires coûteuses. Le CR restait faible.

Nous avons utilisé un nouveau concept de l'augmentation de la taille du bloc de traitement en fonction de la taille de l'image traitée. Les résultats obtenus révèlent une dépendance significative entre la taille de l'image traitée et la taille des sous-images en fonction du CR optimal. Le nombre optimum de neurones de la couche cachée du réseau de neurones dépend de la taille de sous-images en fonction de la qualité visuelle de l'image restaurée.

Nous avons calculé la somme des trois paramètres du réseau de neurones à l'étape de décorrélation de l'image (colonne droite de la figure 5.1). Ce volume de données sert à la restauration de l'image dans le décodeur. Rappelons que ces trois composantes sont : un vecteur des valeurs de neurones de la couche cachée, une matrice des poids synaptiques entre la couche cachée et la couche de sortie et le vecteur des coefficients des seuils des neurones de la couche de sortie.

Ces mesures ont été faites pour différents niveaux de la représentation pyramidale (entre le niveau $p=1$ (512x512), jusqu'à la dernière couche de la pyramide, niveau 10 (1x1)). Les résultats sont présentés au tableau 5.1.

Tableau 5.1 – Structure pyramidale de l'image d'entrée (512x512 pixels, 262 144 Octets) en fonction des paramètres du réseau de neurones à trois couches

Niveau de la pyramide	Nombre de sous images	Taille de sous image	Structure des données	Nombre de neurones des couches du RNN : (m^2-n-m^2) l'entrée : caché : sortie	Données nécessaires à la restauration d'images [Octet]
1	1	512 x 512	1 x 1 (512)	262144 x 512 x 262144	1 075 843 072
2	4	256 x 256	2 x 2 (256)	65536 x 256 x 65536	134 750 208
3	16	128 x 128	4 x 4 (128)	16384 x 128 x 16384	16 924 672
4	64	64 x 64	8 x 8 (64)	4096 x 64 x 4096	2 162 688
5	256	32 x 32	16 x 16 (32)	1024 x 32 x 1024	335 872
6	1024	16 x 16	32 x 32 (16)	256 x 16 x 256	165 888
7	4096	8 x 8	64 x 64 (8)	64 x 8 x 64	266752
8	16384	4 x 4	128 x 128 (4)	16 x 4 x 16	524 928
9	65536	2 x 2	256 x 256 (2)	4 x 2 x 4	1 048 672
10	262144	1 x 1	512 x 512 (1)	1 x 1 x 1	2 097 168

Ces résultats ont été obtenus (Tableau 5.1) à partir d'une image de taille 512 x 512 pixels. Après l'analyse de ces données, la structure optimale recommandée du réseau de neurones est : (256 : 16 : 256). Dans ce cas, il faut diviser l'image originale en 1024 sous-images de taille de 16 x 16 pixels chacun, qui correspond au 6e niveau de la pyramide (colonne gauche du tableau 5.1). Pour ce niveau, le volume du fichier (des paramètres du réseau de neurones nécessaires pour restaurer l'image dans un décodeur) est minimal (165 888 octets) (colonne droite du tableau 5.1). Ce tableau peut être utilisé à titre de recommandations en ce qui a trait au choix de la structure du réseau de neurones pour diverses tailles de l'image traitée.

Ainsi, nous avons étudié la dépendance de la taille des sous-images pour différentes tailles d'images d'entrées (256x256, 512x512, 1024x1024, 2048x2048, 4096x4096 et 8192x8192) en fonction du CR optimal. Nos recommandations à propos du choix de la taille de sous-image et de la structure du réseau de neurones optimum en fonction de la taille de l'image traitée sont présentées ci-dessous (Tableau 5.2).

Tableau 5.2 – Résultats de choix de la taille de sous-image

Taille de l'image traitée	Taille de sous – image	Structure du réseau de neurones : nombre de neurones : couche d'entrée, couche cachée, couche de sortie	Niveau de compression (CR)
256 x 256	8 x 8	64 : 8 : 64	59,43
512 x 512	16 x 16	256 : 16 : 256	61,44
1024 x 1024	16 x 16	256 : 16 : 256	100,72
2048 x 2048	32 x 32	1024 : 32 : 1024	125,39
4096 x 4096	32 x 32	1024 : 32 : 1024	203,11
8192 x 8192	64 x 64	4096 : 64 : 4096	253,36

Remarquons que la taille minimale du volume de la sous-image (c.-à-d. le bloc du traitement optimal en fonction de la taille du fichier et de l'architecture du RN) augmente avec la taille de l'image d'entrée. Cela permet d'augmenter le CR en respectant la qualité visuelle acceptable des images restaurées, tandis que des méthodes courantes sont mal adaptées pour de telles augmentations du bloc de traitement.

5.3. Architecture du réseau de neurones

Les résultats du Tableau 5.2 montrent aussi que la meilleure structure du réseau de neurones est en fonction de la taille de la sous-image choisie. Notons que deux parties de la matrice de coefficients de transformations (les *valeurs de liaisons entre la couche cachée et la sortie ainsi que les valeurs de seuils de la couche sortie*) peuvent être utilisées à côté du décodeur pour restaurer plusieurs (ou milliers) sous-images. Par exemple, nous pouvons définir mille matrices de coefficients de transformations dans le domaine d'intérêt. Après, nous pouvons transférer toutes ces matrices à côté du décodeur, au préalable. Dans ce cas, nous pouvons transférer l'image compressée (des valeurs de neurones de la couche cachée) au décodeur, en ajoutant uniquement un numéro correspondant à la matrice utilisée pour compresser cette image dans le

codeur. Donc, connaître l'architecture du réseau de neurones est important pour gérer adéquatement le processus de la transformation des données.

5.4. Discussion sur des résultats de paramètre SSIM

Le critère de similarité structurelle a montré que les chiffres obtenus par notre méthode sont comparables à ceux obtenus par la méthode JPEG2000. Ainsi, cela signifie que la qualité visuelle devrait aussi être semblable pour ces deux méthodes. Or, en réalité, les images qui ont été compressées par notre nouvelle méthode ont une meilleure qualité visuelle que celles qui ont été compressées par la méthode JPEG2000 (article 4.3).

Rappelons que pour évaluer la qualité visuelle, SSIM divise l'image en sous-blocs de taille 8x8 pixels tandis que nous, nous avons divisé l'image en sous-blocs de taille 16x16 pixels. Cela a probablement changé la relation structurale entre les deux images, avec un bloc de traitement 8x8 pour une image compressée avec la méthode JPEG2000 et un bloc de traitement 16x16 pour une image compressée avec notre méthode. Ceci a donc biaisé l'effet de comparaison entre ces deux méthodes.

5.5. Temps des calculs

Le temps des calculs est un paramètre important en ce qui a trait à la commercialisation prochaine de notre méthode.

Nous avons amélioré le temps des calculs au niveau du codeur en divisant une matrice des vecteurs d'entrées en trois parties (*l'entraînement, la validation et la mise à l'essai*). De plus, nous avons montré l'efficacité d'utilisation d'un seul réseau de neurones pour la série d'images de télédétection.

Ainsi, nous avons testé notre algorithme à partir d'ordinateurs « Mammouth » (Anonyme, 2010), et nous avons obtenu des résultats 5 fois plus rapidement qu'avec un ordinateur ordinaire pour la même quantité de données d'entraînement, en obtenant le même niveau d'erreur. Avec les systèmes multiprocesseurs, nous supposons être en mesure d'éliminer ces limitations.

Au niveau du décodeur le temps de restauration d'image est comparable aux méthodes courantes.

La véritable efficacité de notre méthode s'atteindra grâce à l'utilisation des systèmes multiprocesseurs procédant par calculs parallèles.

5.6. Correspondance au système d'être humain

Il faut souligner que notre approche correspond au système de la vision de l'être humain. Par exemple, pour bien voir des éléments fins de l'image satellitaire étudiée, c.-à-d. des voitures, des piscines, des zones résidentielles, des êtres humains, il faut se rapprocher des objets ciblés. En s'approchant, nous changeons la taille du bloc de traitement par l'œil de l'être humain. Pour notre méthode, nous faisons la même chose, en diminuant la taille du bloc de traitement.

Par contre, pour voir de gros éléments, c.-à-d. des rivières, des lacs, des forêts, il faut s'éloigner (en augmentant la taille du bloc de traitement). Cette propriété est rendue possible par la réduction des corrélations internes des images de THRS, par la modification de la distribution statistique de l'information et par le regroupement des informations similaires.

6. Contribution

6.1. Contribution scientifique

Nous avons proposé une nouvelle approche de représentation des données numériques. La nouvelle méthode AIPD-BPNN présente les données d'images de télédétection selon la représentation pyramidale inverse et adaptative dans l'espace de la couche cachée du réseau de neurones à rétropropagation. Cette méthode hybride et originale est basée sur la décomposition pyramidale adaptative avec l'utilisation de réseaux de neurones permettant d'atteindre la compacité des données en conservant la qualité visuelle de l'image comprimée.

Nous avons comparé les résultats obtenus pour une méthode AIPD-BPNN avec les résultats des méthodes courantes telles que JPEG, JPEG2000 et LuraWave. Les résultats de l'analyse comparative montrent une haute qualité visuelle des images de THRS restaurées par la méthode AIPD-BPNN. Ainsi, les hypothèses de recherche ont été vérifiées et confirmées.

6.2. Contribution scientifique appliquée

Nous avons estimé le potentiel d'application de la méthode AIPD-BPNN pour des images testées, images en couleur, des images de télédétection, des images de THRS et une transmission progressive des images. Nous avons défini une approche pour l'accélération du temps d'apprentissage de l'algorithme par voie du traitement parallèle de l'information.

En se basant sur nos résultats expérimentaux, nous avons confirmé que pour la transformation de l'image par la méthode AIPD-BPNN, la distribution de l'erreur est mieux répartie (que pour JPEG et JPEG2000) et qu'elle est moins visible pour l'œil d'un être humain. Si l'erreur est concentrée à des places définies (souvent cela se passe à la frontière des contours), il est plus facile de la remarquer et cela diminue la qualité visuelle de l'image. Pour la transformation en ondelettes, nous avons visuellement remarqué le « flou » des contours.

7. Originalité du projet

La nouveauté de l'approche hybride de représentation adaptative d'images visuelles de télédétection de THRS réside dans le fait que nous utilisons des performances de la décomposition pyramidale inverse en réalisant des calculs à l'aide des réseaux de neurones. Cette combinaison permet de réaliser efficacement la transformation progressive d'images de télédétection. La modélisation se passe à chaque niveau de la pyramide. Les calculs se passent séparément pour chaque niveau de la pyramide. Les différentes couches de la pyramide correspondent à différentes estimations de la résolution de l'image. Cela permet de cibler l'accès de l'information en fonction des exigences des utilisateurs (la qualité de l'image ou le temps de transfert en éliminant des calculs inutiles).

Le traitement par réseau de neurones permet de trouver l'opérateur optimum du regroupement des pixels. Les coefficients du réseau de neurones se calculent en fonction du contenu des images d'entrées du domaine d'étude. Les coefficients de la transformation reçus permettent d'obtenir une qualité visuelle supérieure des images restaurées, comparée aux méthodes classiques.

En combinant deux méthodes de traitement, nous avons obtenu une qualité visuelle comparable pour des images compressées pour le CR 100 fois. En jouant avec la taille du bloc de traitement, nous avons montré que la qualité visuelle était supérieure, en comparaison des méthodes courantes pour le CR 247 fois. Ceci a surpassé les attentes que nous avions au départ.

8. Conclusions générales

L'objectif principal, ainsi que les objectifs spécifiques, de ce projet de recherche ont été atteints avec succès. Selon les résultats approuvés et publiés par les évaluateurs des revues à comité de lecture, nous sommes maintenant assurés que l'algorithme hybride à la base de la représentation pyramidale adaptative des images de THRS par pyramide inverse avec des réseaux de neurones permet de transformer et de décoder les images de télédétection avec une qualité visuelle comparable à ce que des algorithmes actuels proposent dans le domaine du traitement des images numériques.

De plus, les résultats reçus pour de hauts niveaux de la compression ($CR=200$ et plus) des images de télédétection montrent l'efficacité de la méthode hybride adaptative proposée. La qualité visuelle des images restaurées par notre méthode surpasse la qualité visuelle des images restaurées par les algorithmes JPEG et JPEG2000, pour des niveaux identiques de compression.

Nous avons confirmé la performance significative de notre méthode en utilisant la transformation par réseau de neurones, comparativement à la transformée discrète par cosinus pour le format JPEG. Nous confirmons la limite de l'algorithme JPEG pour des images de THRS. L'algorithme JPEG donne une qualité visuelle acceptable des images restaurées jusqu'au niveau de la compression $CR = 60$. Avec de grands CR, l'algorithme JPEG montre des résultats médiocres. L'algorithme de compression par ondelettes (JPEG2000) permet d'atteindre un CR de l'ordre de 247, mais l'image restaurée contient des zones 'floues'.

Considérant l'ensemble des résultats obtenus, tant par les analyses qualitatives que quantitatives, l'algorithme de compression développé est satisfaisant pour la compression d'images de télédétection de THRS, vis-à-vis des pertes engendrées. La maîtrise des pertes devrait passer par une série d'indicateurs rendant compte de la fidélité de l'image compressée par rapport à l'information originale. Cependant, les logiciels commerciaux de compression actuels (MrSid et ECW) ne proposent pas de tels indicateurs. De surcroît, ils fonctionnent selon le principe d'une boîte noire ne donnant aucune indication sur les traitements appliqués, hormis le fait qu'il exploite une compression par ondelettes.

8.1. Avantages

Les résultats obtenus dans ce travail confirment l'hypothèse que des images satellites de haute définition, des images multispectrales et des images hyperspectrales possèdent un haut degré de redondances. Cela permet d'utiliser efficacement la méthode AIPR-BPNN pour la compression d'images de haute définition dans les applications de télédétection, en offrant une qualité visuelle comparable ou meilleure, toujours en comparaison avec les méthodes courantes.

L'utilisation du réseau de neurones est bien adaptée à l'augmentation du volume des images satellitaires, à l'archivage et au stockage de ce type de données.

La nouvelle méthode offre de larges opportunités pour des domaines d'application de télédétection : le traitement d'images numériques, en traitant l'image compressée avec le volume abordable au lieu de traiter l'image originale avec un gros volume; le transfert progressif de l'image via Internet; la recherche dans des bases de données d'images, etc.

8.2. Limitations

Notre méthode AIPR-BPNN est asymétrique (le codeur est plus complexe que le décodeur) et ceci représente une limitation aux domaines d'applications qui exigent le traitement en temps réel, c'est-à-dire des applications pour lesquelles le temps d'entraînement est crucial.

Pour l'instant les vrais calculs parallèles sont limités par l'utilisation des ordinateurs spéciaux, par exemple « Mammouth » de l'Université de Sherbrooke (Anonyme, 2010), qui exigent un accès spécial, ainsi qu'un logiciel spécial. Deuxièmement, la technologie de fabrication des processeurs ne permet pas de produire la carte de processeurs de plus de quatre cœurs pour les ordinateurs ordinaires (c.-à-d., le portable ou l'ordinateur au bureau). Cela limite la supériorité véritable des réseaux de neurones dans les calculs parallèles, particulièrement pour le mode d'apprentissage du RN.

9. Perspective

Le développement de nouvelles modalités d'imagerie de THRS et d'une exploitation efficace des modalités existantes ne manque pas de poser de nouveaux défis à la communauté :

- a. gestion et analyse de données hyperspectrales (plus d'une centaine de canaux);
- b. fusion de grandes masses de données numériques spatiales et temporelles;
- c. n-dimensionnels (3D, 4D avec le spectre, 5D avec l'évolution temporelle, 6D avec la polarisation, etc.);
- d. image en stéréo, 3D (voir l'annexe 6);
- e. analyse et interprétation de données par des algorithmes avec des réseaux de neurones;
- f. utilisation des cartes des processeurs avec un nombre de processeurs plus grand;
- g. transmission des données compressées à partir de satellites.

La prise en compte de la complexité et de la variabilité croissante apportée par les systèmes imageurs récents rend ainsi nécessaire le développement de nouveaux paradigmes de traitement associant modèles, représentations, algorithmes et architectures innovants. Les besoins s'expriment tout autant pour des traitements génériques 'classiques' (codage et compression, amélioration, restauration et déconvolution, recalage et fusion, détection, segmentation, suivi temporel, reconnaissance et interprétation) que pour des traitements spécifiques liés à chaque domaine thématique (astronomie, télédétection, imagerie médicale, etc.). Dans de nombreux cas, l'automatisme et la reproductibilité des traitements restent un enjeu majeur.

9.1. Nanotechnologie et calculs parallèles

En prévoyant les changements des méthodes de fabrication des processeurs (nanotechnologie, actuel 45nm vers 32 nm en futur) (Intell, 2008) et le changement de la capacité du système de traitement par voie de l'augmentation de la quantité de processeurs (multicoeurs, actuel 2 cœurs), on peut supposer que la question sur les systèmes parallèles du traitement de l'information est extraordinairement actuelle. Notre méthode est bien préparée à ces changements technologiques, puisque les

réseaux de neurones sont naturellement destinés aux calculs distribués et aux calculs parallèles.

9.2. Canal de transmission des données numériques

En vue de l'augmentation de la performance des nouvelles technologies d'acquisition d'information, les canaux de transmission restent toujours un point faible, malgré les technologies récentes, telles que la fibre optique. La proportion inégale entre l'augmentation des données et la capacité du canal de transmission laisse ainsi le vrai problème de la recherche des algorithmes de la transformation des données numériques dans le futur.

L'importance des données de télédétection, telles que hors du spectre visible, sont essentielles parce qu'elles offrent la possibilité de mieux comprendre le monde réel et d'utiliser ces connaissances pour l'humanité de différentes façons. Notre méthode est le moyen pour améliorer le traitement des données numériques géospatiales avec le but de simplifier l'utilisation des logiciels tels que SIG pour la population.

10. Références

- Anonyme (1991) Rapport de synthèse du Groupe Réseaux de neurones. Observatoire français des techniques avancées. Groupe Réseaux de neurones 1991, 127 p.
- Anonyme (2010) Réseau québécois de calcul de haute performance. <https://rqchp.ca/?pageId=566>
- Adelson, E.H. and Simoncelli, E. (1981) Orthogonal pyramid transforms for image coding, Visual Communications and Image Processing II, Proc. SPIE, Vol.845, pp.50-58, 1981.
- Ameur, S., Adane A-E-H. et Lahdir M. (2002) Compression d'images MÉTÉOSAT en sous-bandes par transformation discrète en cosinus et quantification vectorielle / *Téledétection*, 2002, vol. 2, n° 4, p. 255.
- Antonin, M. (2000) Compression d'images fixe selon le contenu. Mémoire de maîtrise en science appliquées. Université de Sherbrooke, 2000.
- Alföldi, T. T. (1978) Introduction aux images numériques et aux techniques d'analyse numérique. Ottawa : Centre canadien de télédétection, Énergie, mines et ressources, 1978.
- Aizzi, B., Alparone L., Baronti S. and CNR F. (1996) A reduced laplacian for lossless and progressive image communication, IEEE transaction on communication, vol. 44, no. 1, pp. 18-22, January 1996.
- Barlaud, M. et Labit, C. (2002) Compression et codage des images et des vidéos. Paris : Hermes Science Publications, 2002, 319 p.
- Bloch, I. and Maitre H. (1994) Fusion de données en traitement d'images : modèles d'information et décisions. Traitement du signal. Vol. 11(6), pp. 435-446.
- Burger, W. and Burge M. J. (2009) Principals of digital image processing: core algorithms. London: Springer, c2009, 327 p.
- Burt, P. J. and Adelson E.H. (1983) The Laplacian pyramid as a compact image code. IEEE Trans.Commun. COM-31, p. 532-540.
- Caloz, R. et Collet C. (2001) Précis de télédétection. Volume 3, Traitements numériques d'images de télédétection. Sainte-Foy : Presse de l'Université de Québec, 2001, 386 p.
- Campbell, J. B. (2002) Introduction to remote sensing. New York : Guilford Press, c2002, 621 p.
- CARTEL (2009) Cartographie de la couverture terrestre de l'OTDD : méthode, validation, résultats et applications. Conférence organisée par le CARTEL. Présentée par Stephen Côté, Ressources naturelles Canada, Service canadienne des forêts, Centre de Foresterie des Laurentides, 27 février 2009.
- Cohen, A. (1992) Ondelettes et traitement numérique du signal. Paris : Masson, 1992, 205 p.
- Chen, T. M., Staelin D. H. and Arps R.B. (1987) Information content analysis of Landsat image data for compression. IEEE Transactions on Geoscience and Remote sensing, Vol. GE-25, No 4, pp. 499-501, July 1987.
- Chen, C.H. and Wang, P.S.P. (2005) Handbook of pattern recognition and computer vision. Singapore: Hackensack, NJ: World Scientific, c2005, 639 p.
- Chen, C.W. and Zhang, Ya, Q. (1999) Visual information representation, communication, and image processing. New York:Marcel Dekker, c1999, 551 p.
- Dasarathy, B. V. (1995) Image data compression: block truncation coding. Los Alamitos, Calif.: IEEE Computer Society Press, 1995, 284 p.
- Desjardins, R. (2000) La télédétection: perspective analytique. Paris : Édition ESTEM; Montréal : AUPELF-UREF, c2000, 226 p.

- FAQ (2007) <http://www.faqs.org/faqs/ai-faq/neural-nets/part1/preamble.html>
- Gelenbe, E., Sungur M. et Cramer Ch. (1996) Learning random networks for compression of still and moving images. Christopher E Cramer.
- GeoEye, (2006) Image satellite Ikonos 2. Sherbrooke (Québec). DVD-ROM. Données matricielles. Cartothèque de l'Université de Sherbrooke.
- GeoEye, (2007)
http://www.geoeye.com/whitepapers_pdfs/GcteoEye_IKONOS_Produ_Guide_v17.pdf
- Gagnon, L. et Lalonde M. (1999) Rapport de veille stratégique. Vision artificielle et traitement d'image. Centre de recherche informatique de Montréal Bibliothèque nationale du Québec, 1999, 138 p.
- Gibson, P. J. and Power C. H. (2000) Introductory remote sensing: digital image processing and applications. London: Routledge, 2000, 249 p.
- Gomarasca, M.A. (2009) Basics of geomatics. Dordrecht : Pays-Bas: Springer, c2009, 656 p.
- Gonzalez, R.C. and Woods, R.E. (2008) Digital image processing. Harlow: Prentice Hall, c2008, 954 p.
- Groupe Réseaux de neurones (1991) Rapport de synthèse du Groupe Réseaux de neurones. Observatoire français des techniques avancées, 1991.
- Guillois, J.-P. (1996) Techniques de compression des images. Paris : Hermès, c1996, 252 p.
- Gutter, H. (1995) La compression des images numériques. Conservatoire national des arts et métiers, Centre Associé de Clermont-Ferrand. Paris, Hermès, c1995, 79 p.
- Held, G. and Marshall T.R (1996) Data and image compression: tools and techniques, 4th ed., Wiley UK, Chichester.
- Hérault, J. (1994) Réseaux neuronaux et traitement du signal. Paris :Hermès, c1994, 313 p.
- Hoggar, S. G. (2006) Mathematics of digital images: creation, compression, restoration, recognition. Cambridge, UK: Cambridge University Press, 2006, 854 p.
- Hu, Y. H. and Hwang J.-N. (2002) Handbook of neural network signal processing. Boca Raton: CRC Press, c2002.
- Hubel, D. H. (1988) Eye, Brain, and Vision. Scientific American Library, c1988, 256 p.
- Hwang, W.-J. and Derin H. (1995) Multiresolution multiresource progressive image transmission, IEEE transaction on image processing , vol. 4, no. 8, page 1128-1140, August.
- Intell (2008). Intel 32nm Logic Technology http://www.intel.com/technology/architecture-silicon/32nm/index.htm?iid=tech_arch_45nm+rhc_32nm
- Isar, A., Cubitchi A. and Naforita M. (2002) Algorithmes et techniques de compression. AUPELE-UREF, AUF – Agence Université de la francophonie, 2000.
- Jensen, J. R. (2005) Introductory digital image processing: a remote sensing perspective. Upper Saddle River, N.J.: Prentice Hall, 2005, 526 p.
- Jolion, J.-M. (2001) Les systèmes de vision. Paris : Hermès Science Publications, c2001, 361 p.
- Kalyani, D. V. (2009) Pattern recognition using neural and functional networks. Berlin: Springer, c2009, 226 p.
- Karam, L. J. (2000) Lossless Coding. p. 461-474. in A. Bovik (éd.); Handbook of Image and Video Processing. Academic Press, London, 891 p.

- Kim, S. (1991) *The representational theory of mind : an introduction*. Oxford, OX, UK; Cambridge, Mass., USA: B. Blackwell, 1991, 252 p.
- Kou, W. (1995) *Digital image compression: algorithms and standards*. Boston:Kluwer Academic Publishers, c1995, 192 p.
- Kountchev, R., Milanova M. G., Ford C., Rubin S. H., Kountcheva R. (2004) Lossless Image Compression with IDP and Adaptive RLC. CISST 2004, pp. 608-612.
- Kountchev, R. et Rubin S. (2000) *Image Compression with Adaptive Invers Difference Pyramide*. Space and Naval Warfare Systems Center, San Diego, USA.
- Kovacs, M. and Ranganathan N. (1995) Jaguar: A fully pipelined VLSI architecture for JPEG image compression standard. *Proceedings of IEEE*, vol. 83, n° 2, p. 247-258.
- Kunt, M. (1996) *Video coding : the second generation approach*. Boston: Kluwer Academic, 1996, 433 p.
- Kunt, M. (1984) *Traitement numérique des signaux*. Lausanne : Presses Polytechniques romandes, c1984, 402 p.
- Luo, F.-L. (1997) *Applied neural networks for signal processing*. New York: Cambridge University Press, 1997, 367 p.
- Mallat, S.G. (1989) A theory for multiresolution signal decomposition: The wavelet transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, p. 674-693.
- Marsault, X. (1995) *Compression et cryptage des données multimédias*. Paris : Hermès, 1995, 243 p.
- Moulin, P. (2000) Multiscale Image Decompositions and Wavelets. p. 289-300, in A. Bovik (réd.) *Handbook of Image and Video Processing*. Academic Press, London, 891 p.
- Perry, S.W., Guang L. and Wong H.-S. (2002). *Adaptive image processing: a computational intelligence perspective*. Boca Raton, Fla.: CRC Press, c2002, 272 p.
- Pointet, A., Caloz, R., et Riedo, M. (2003) *Compression des images de télédétection*. Laboratoire de systèmes d'information géographique. École polytechnique fédérale de Lausanne. *Télédétection*, 2003, vol. 3, n° 2-3-4, p. 323-335.
- Prasard, B. and Prasanna S.R.M. (2008) *Speech, Audio, Image and Biomedical Signal Processing using Neural networks*. Berlin: Springer, c2008.
- Provencher, L. et Dubois J.-M. M. (2007) *Précis de télédétection*. Volume 4. Méthodes de photointerprétation et d'interprétation d'image. Presse de l'Université de Québec, 2007, 508 p.
- Rao, K.R. et Yip P.C. (2001). *The transform and data compression handbook*. Boca Raton, Fla: CRC Press., c2001.
- Read, R. and Graham, R. (2002) *Manual of aerial survey: Primary data acquisition*. Whittle Publishing (CRC Press), Boca Raton, 408 p.
- Saad, E. W. and Wunsch D. C. (2007) Neural network explanation using inversion / Emad W. Saada, Donald C. Wunsch. *ScienceDirect, Neural Networks* 20 (2007) p.78-93.
- Salomon, D. (2004) *Data compression*. New York: Springer, 2004.
- Steinberg, J. (2000) *Cartographie : télédétection, systèmes d'information géographique*. Paris : SEDES, 2000, 159 p.
- Shannon, C.E. (1948) A mathematical theory of communication, *Bell System Technical Journal*. Vol.27, pp.379-423 and 623-656, July and October, 1948.

- Shannon, C.E. and Weaver, W. (1963) A mathematical theory of communication University of Illinois Press, Urbana.
- Schalkoff, R.J. (1992) Pattern Recognition; Statistical, Structural and Neural Approaches. New York : John Wiley & Sons, c1992, 364 p.
- Tonye, E. (2000) Le traitement des images de télédétection par l'exemple. Paris : Gordon and Breach Science Publishers, c2000, 179 p.
- Topiwala, P. N. (1998) Wavelet image and video compression. Boston: Kluwer Academic Publishers, c1998, 438 p.
- Touzet, C. (1992) Les réseaux de neurones artificiels: Introduction au connexionnisme. Paris, 1992, 127 p.
- Tzagaris, V., Anastassopoulos V. and Lampropoulos G.A. (2005) Fusion of hyperspectral data using segmented PCT for color representation and classification. IEEE Transactions on geoscience and remote sensing. Vol. 43, No 10, pp. 2365-2375, Octobre 2005.
- Tzovaras, D. and Strintzis M. (2002) Optimal Constraction on Reduced Pyramids for Lossless and Progressive Image Coding. IEEE Trans. on circuits and Systems-II Analog and Digital Signal Processing, Vol. 47, no 4, April 2000, pp. 332-348.
- Wang, Z. and Bovik A.C. (2009) Mean squared error: Love it or leave it? IEEE Signal Processing magazine, pp.98-117, January 2009.
- Wang, Z., Bovik A.C., Sheikh H.R. and Simoncelli E.P. (2004) Image quality assessment: From error visibility to structural similarity, "IEEE Transactions on Image Processing", Vol. 13, no. 4, pp. 600-612, april 2004.
- Zharkova, V. and Jain L. C. (2007) Artificial intelligence in recognition and classification of astrophysical and medical images. Berli:New York:Springer, c2007, 374 p.

Annexe 1. Caractéristiques principales du radiomètre équipant le satellite Ikonos

- Satellite Ikonos : L'instrumentation embarquée est destinée à l'acquisition d'images à haute résolution. Le radiomètre permet ainsi d'offrir des images couvrant un champ d'observation de 11 km * 11 km, avec une résolution spatiale de 1 m en mode panchromatique et de 4 m en mode multispectral. Pour l'année 2007 les images du domaine complet de 275 millions de kilomètres carrés étaient reçues.

	Bits per Pixel	Number of Bands	Resolution	File Size per Square Kilometer
Black & White	8	1	1-meter	1 megabyte
	11	1	1-meter	2 megabyte
Multispectral (true or false color)	8	3	4-meter	1.875 megabyte
	11	3	4-meter	3.375 megabyte
Multispectral (4-band)	8	4	4-meter	2.5 megabyte
	11	4	4-meter	4.5 megabyte
Color (true or false color)	8	3	1-meter	3 megabytes
	11	3	1-meter	6 megabytes
Color (4-band)	8	4	1-meter	4 megabytes
	11	4	1-meter	8 megabytes
Bundle (true or false color)	8	4	1-meter + 4-meter	1.1875 megabytes
	11	4	1-meter + 4-meter	2.375 megabytes
Bundle (4-band)	8	5	1-meter + 4-meter	1.25 megabytes
	11	5	1-meter + 4-meter	2.5 megabytes

Figure 1 – Caractéristiques principales du radiomètre équipant le satellite Ikonos (GeoEye, 2007)

La mode panchromatique et tous quatre modes multispectraux ont une gamme de 11 bits dynamique. Avec la décision de 11 bits, les détails sous les ombres, les moments principaux, et les scènes basses contrastées peuvent plus facilement être distingués que dans les représentations de 8 bits.

L'octet (byte en anglais) est une unité de mesure en informatique mesurant la quantité de données. Un octet est lui-même composé de 8 bits, soit 8 chiffres binaires.

Multiples d'octets tels que définis par IEC 60027-2

Kiloctet	(ko)	–	10^3
Mégaoctet	(Mo)	–	10^6
Gigaoctet	(Go)	–	10^9
Téraoctet	(To)	–	10^{12}
Pétaoctet	(Po)	–	10^{15}
Exaoctet	(Eo)	–	10^{18}
Zettaoctet	(Zo)	–	10^{21}
Yottaoctet	(Yo)	–	10^{24}

Selon la Commission électrotechnique internationale (CEI) ou *International Electrotechnical Commission (IEC)* en anglais.

Annexe 2. Étapes de traitement d'images numériques

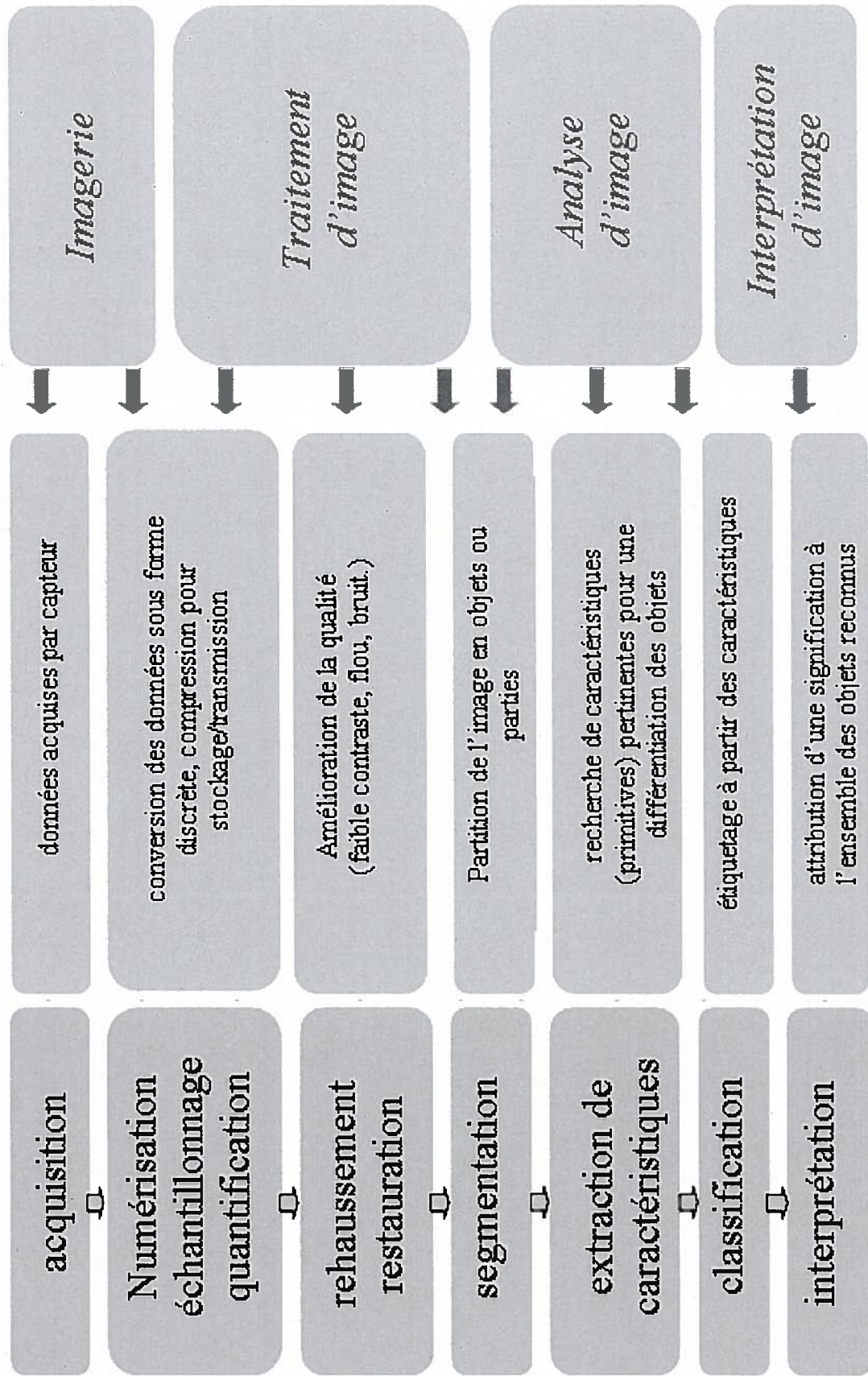


Figure 1 – Étapes principales dans un système de traitement d'images (Guillois, 1996)

Annexe 3. Vision artificielle et traitement d'images

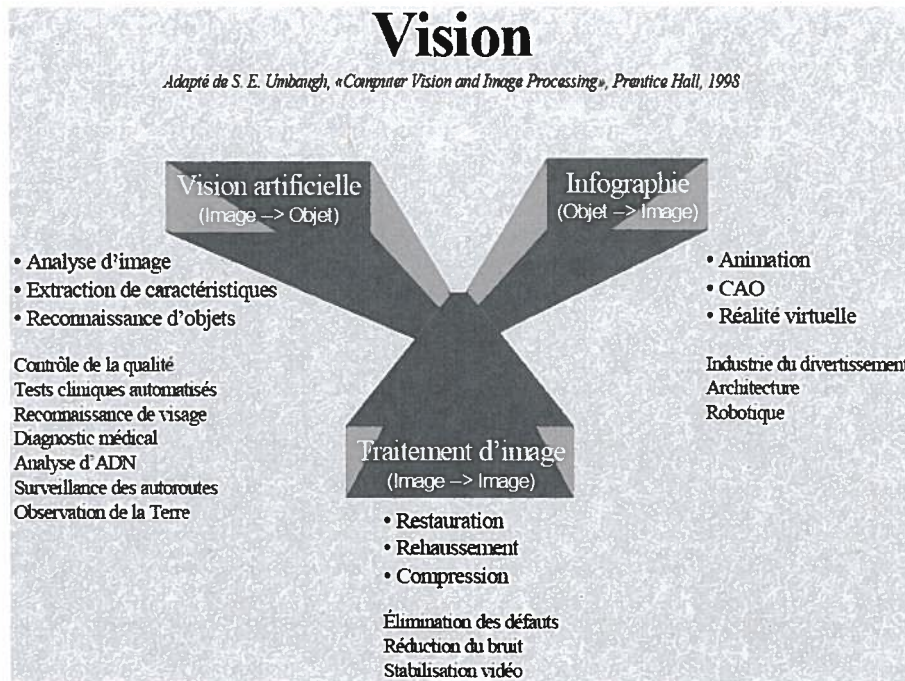


Figure 1 – Relations des sujets dans le domaine de traitement d'images numériques (Gagnon et Lalonde, 2000, p.13)

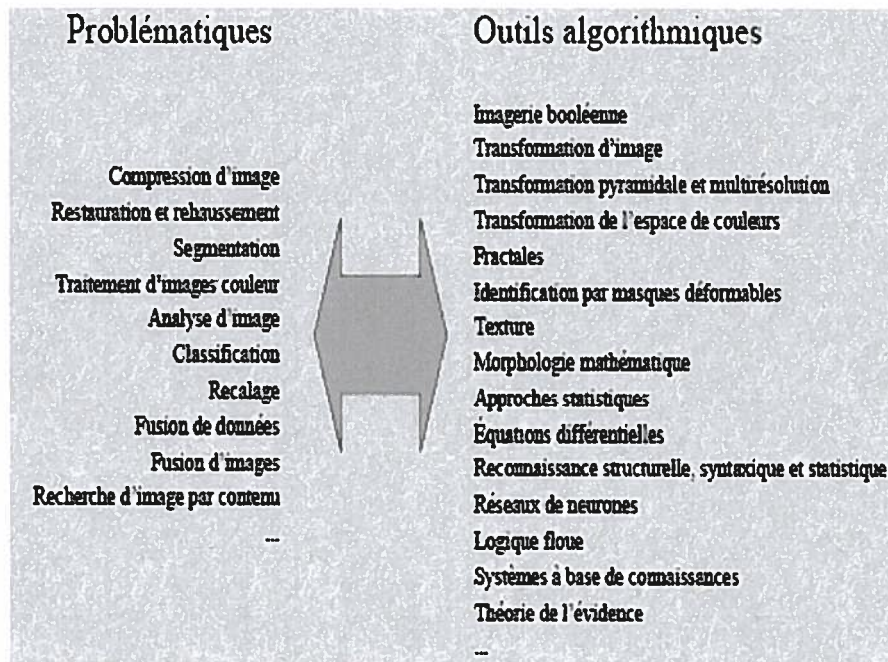


Figure 2 – Problématiques et des outils algorithmiques du domaine de traitement d'images numériques (Gagnon et Lalonde, 2000, p.16)

Annexe 4. Système de la vision d'être humain (1/3)

La vision humaine

Le point de départ de la vision humaine est la lumière réfléchiée ou émise par les objets, laquelle est captée par l'œil et transmise au cerveau.

La vision humaine est un phénomène particulièrement complexe qui relève non seulement de l'anatomie de l'œil et de l'optique, mais également de la physiologie et de la psychologie du sujet. Les facteurs liés à l'anatomie, à l'optique et la physiologie peuvent se produire artificiellement. En revanche, les facteurs psychologiques sont propres à l'humain. Ils sont acquis par l'expérience, ce qui les rend difficiles à reproduire artificiellement, compte tenu des moyens électronique disponibles présentement (Provencher et Dubois, 2007).

L'interprétation d'image se basera sur les critères de teinte, de brillance, de forme, de texture, d'arrangement, de taille et de contexte; il sera toujours difficile de quantifier plusieurs de ces variables; c'est pourquoi l'interprétation visuelle deviendra l'élément fondamental dans le processus d'extraction d'information (Desjardins, 2000).

Description du système visuel

L'œil

- La couche sensible : **la rétine**.
- La convergence du rayonnement lumineux incident est due à la forme convexe de la cornée mais également à la lentille accommodatrice : **le cristallin**.
- L'adaptabilité à la lumière ambiante est réalisée par **la pupille** qui se comporte comme un diaphragme.
- **Les muscles oculomoteurs** commandent l'orientation de l'axe optique, donc le mouvement des yeux.

La rétine

- Zone la plus sensible de la rétine : la **FOVEA**
→ environ 1 mm de diamètre = 1 d° de vision angulaire.
- Il existe un second creux dans la rétine : la **papille**
→ départ du faisceau du nerf optique = tâche aveugle.

Annexe 4. Système de la vision d'être humain (2/3)

Les cellules de la rétine

- Les bâtonnets (110 à 125 millions)

→ pour vision scotopique ou vision nocturne.

- Les cônes (4 à 7 millions)

→ pour vision photopique ou diurne.

→ 3 types contenant chacun un pigment visuel, d'absorption spectrale située respectivement dans les parties B, V, R du domaine visible (longueur d'onde : 440, 530, 570 nm).

La distribution des cônes et des bâtonnets sur la rétine est présentée sur la Figure 1.

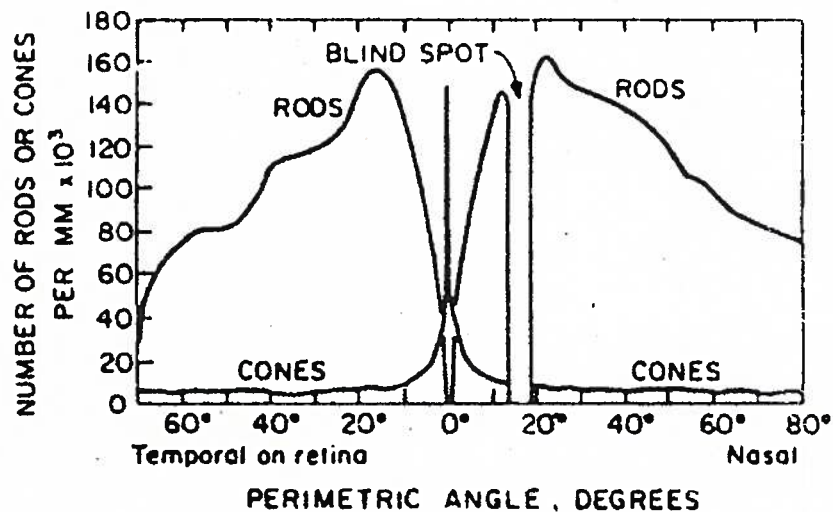


Figure 1 – Distribution des cônes et bâtonnets sur la rétine (Chen et Wang, 2005)

Annexe 4. Système de la vision d'être humain (3/3)

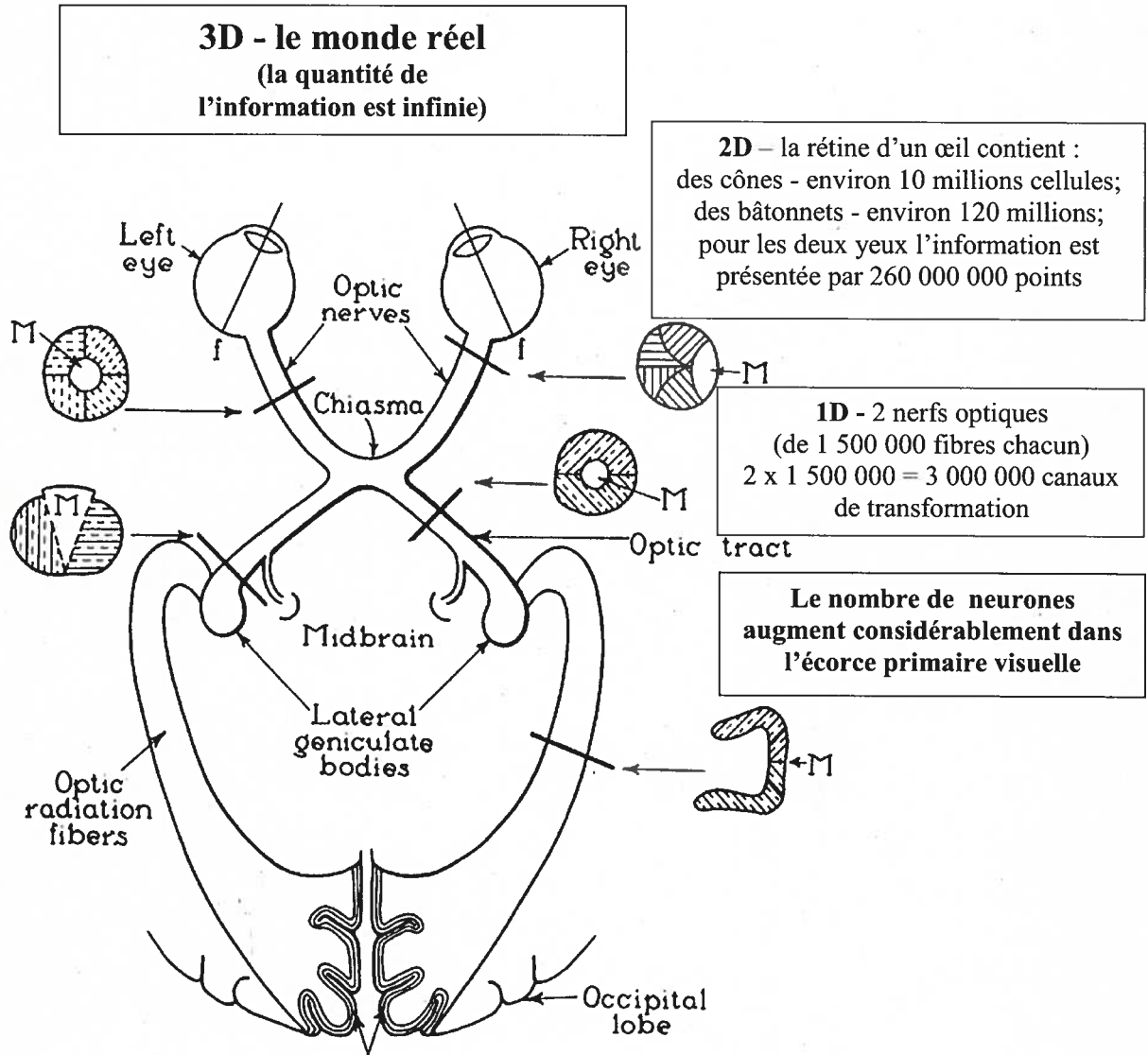


Figure 2 – Système de la vision d'être humain avec des différents niveaux de la transformation du signal optique de 3D (D- dimension) du monde réel à la 1D dans les nerfs optiques (Jolin, 2001)

Annexe 5. Réseau de neurones artificiels (1/4)

Réseau de neurones artificiels

Les réseaux de neurones artificiels consistent en un ensemble d'outils de calcul non-linéaires basés sur le concept d'apprentissage. Ils sont utilisés pour construire des modèles de processus trop complexes que pour être appréhendés d'une façon physique, ou pour analyser des données dont le processus générateur sous-jacent est inconnu.

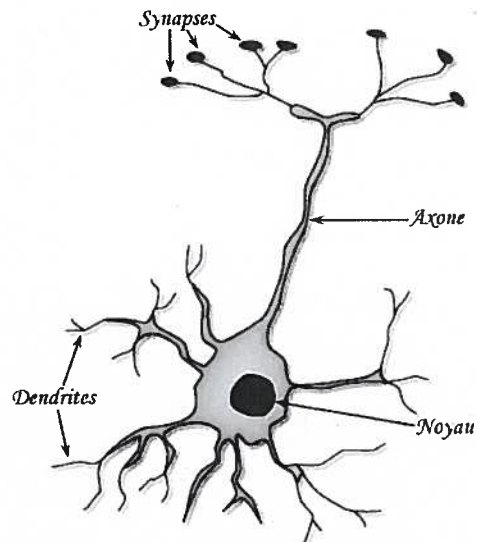


Figure 1 – Neurone biologique (Anonyme, 1991)

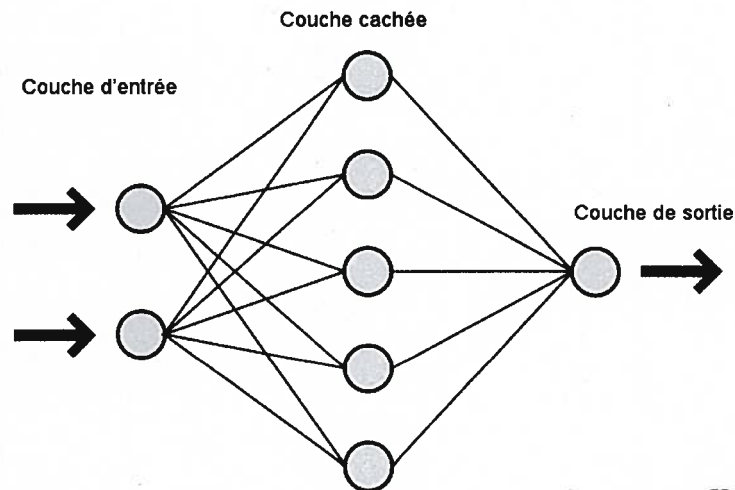


Figure 2 – Réseau de neurones artificiels (Anonyme, 1991)

Annexe 5. Réseau de neurones artificiels (2/4)

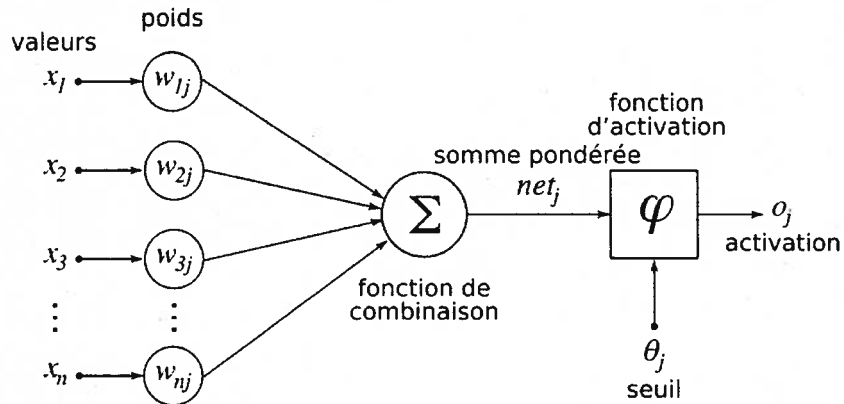


Figure 3 – Structure d'un neurone artificiel (Touzet, 1992)

Les réseaux de neurones artificiels sont utilisés dans des domaines aussi divers que le traitement d'information, la classification (pattern recognition) de données, d'images, de parole, la prédiction de séries temporelles ou encore le contrôle. La plupart des réseaux de neurones artificiels sont des méthodes adaptatives, aptes à modifier le modèle construit en fonction d'un environnement changeant au cours du temps.

Les principaux modèles algorithmiques d'analyse non-linéaire appelés "réseaux de neurones artificiels" (mais non exhaustif) sont (FAQ, 2007):

- les modèles à une couche (adaline, perceptron, mémoire associative) ;
- les perceptrons multi-couches (MLP) ;
- les réseaux à fonctions radiales de base (RBFN) ;
- les modèles de quantifications vectorielles (Lloyd, LVQ, etc.) ;
- l'auto-organisation (cartes de Kohonen).

Réseaux multicouches (MLP)

Apparus en 1985, les réseaux multicouches sont aujourd'hui les modèles les plus employés. Plusieurs couches de traitement leur permettent de réaliser des associations non linéaires entre l'entrée et la sortie. Ils sont ainsi capables de résoudre le cas du "ou exclusif" (Perceptron) (Hu et Hwang, 2002).

On sait depuis les années soixantes que les possibilités de traitement des réseaux multicouches sont supérieures à celle du Perceptron, cependant l'algorithme d'apprentissage manquait. Pour la couche de sortie, on peut appliquer l'apprentissage du Perceptron, mais comment modifier les poids pour les connexions qui ne sont pas en relation avec un neurone de sortie ?

Le problème est ramené à l'obtention d'une estimation de la valeur désirée pour chaque neurone de la couche cachée. La rétropropagation de gradient est une solution à ce problème. Cet algorithme a été proposé indépendamment par trois équipes en 1985. Des recherches bibliographiques ont montré qu'il s'agit en fait d'une redécouverte.

Annexe 5. Réseau de neurones artificiels (3/4)

Le principe utilisé par la rétropropagation ("*backpropagation*" en anglais) de gradient est la minimisation d'une fonction dépendante de l'erreur. Il s'agit d'une méthode générale, largement employée dans d'autres domaines tels que la physique. Une perception intuitive de cet algorithme consiste à considérer l'apprentissage comme la recherche sur la surface de coût de la position de coût minimal. A chaque configuration de poids correspond un coût. Le gradient est une estimation locale de la pente de la surface. La minimisation du gradient permet de parcourir cette surface orthogonalement aux courbes de niveau d'un pas fixé. Les problèmes rencontrés durant l'apprentissage résultent des zones très plates et des minima locaux.

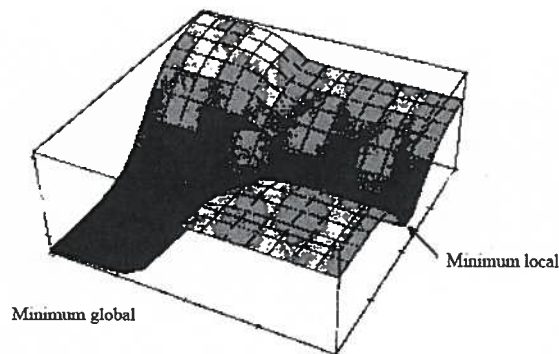


Figure 4 – Recherche de la zone de coût minimal dans l'espace de configuration des poids du réseau pour les exemples de la base d'apprentissage (Hu et Hwang, 2002)

La fonction de transfert (d'activation) de neurone

Les neurones sont continus. La fonction de transfert est une sigmoïde qui peut être définie, par exemple, par l'équation :

$$f(x) = \frac{1}{1 + e^{-\Theta x}} \quad (1)$$

où Θ spécifie la pente de la sigmoïde.

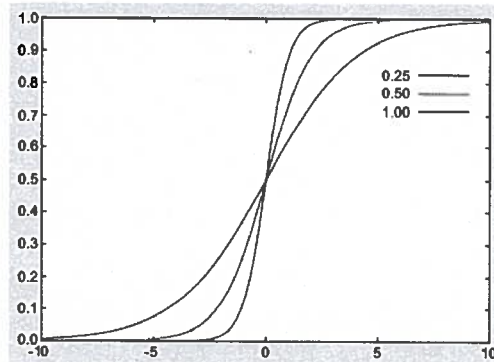


Figure 5 – Graphique illustrant la fonction d'activation sigmoïde pour une pente de 0,25, 0,50 et 1,00

Annexe 5. Réseau de neurones artificiels (4/4)

Apprentissage supervisé avec l'algorithme de la rétropropagation

L'apprentissage est supervisé : on associe une configuration d'entrée à une configuration de sortie. L'algorithme de la rétropropagation est un algorithme de gradient itératif conçu pour minimiser un critère quadratique ("à la puissance 2") d'erreur entre la sortie obtenue d'un réseau multicouche et la sortie désirée.

Cette minimisation est réalisée par une configuration des poids adéquate. L'erreur (e) est la différence entre la valeur désirée (d) pour le neurone de sortie et sa valeur calculée par propagation (x). Ce signal d'erreur permet de définir une fonction de coût :

$$C(W) = M[C_1(W)] = M\left[\sum_j e_{lj}^2(W)\right] \text{ avec } e_{lj} = (d_{lj} - x_{lj}) \quad (2)$$

où,

j - indique un numéro d'indice pour les neurones de sortie et
 l - indique un exemple d'apprentissage.

M est l'opérateur de moyennage, c'est une estimation de la moyenne temporelle dans le cas stochastique. On réalise donc la moyenne des erreurs obtenues pour chacun des exemples de la base d'apprentissage.

Cet algorithme nécessite une fonction continue, non-linéaire et différentiable comme fonction de transfert du neurone :

1. Initialisation des poids à des valeurs aléatoires de faible grandeur;
2. Sélection d'un exemple d'apprentissage (\mathbf{E} , \mathbf{d}) dans la base d'apprentissage ;
3. Présentation de la forme d'entrée (\mathbf{E}) sur la couche d'entrée du réseau;
4. Calcul par propagation de la sortie obtenue (\mathbf{O});
5. Si erreur en sortie alors pour tous les neurones i (depuis la sortie jusqu'à l'entrée) :

Si i est un neurone de sortie alors $y_i = 2 f'(a_i) * (d_i - x_i)$;

Si i est un neurone caché (ou d'entrée) alors $y_i = f'(a_i) * \sum_k (w_{ki} * y_k)$;

(k : neurones compris entre la couche actuelle et la couche de sortie)

6. Application de la procédure de gradient μ est un gain fixé par l'utilisateur.

$$w_{ij}(t+1) = w_{ij}(t) + \mu * y_i * x_j ;$$

7. Tant que l'erreur est trop importante, retour à l'étape 2 (exemple suivant).

Annexe 6. Stéréo satellite image

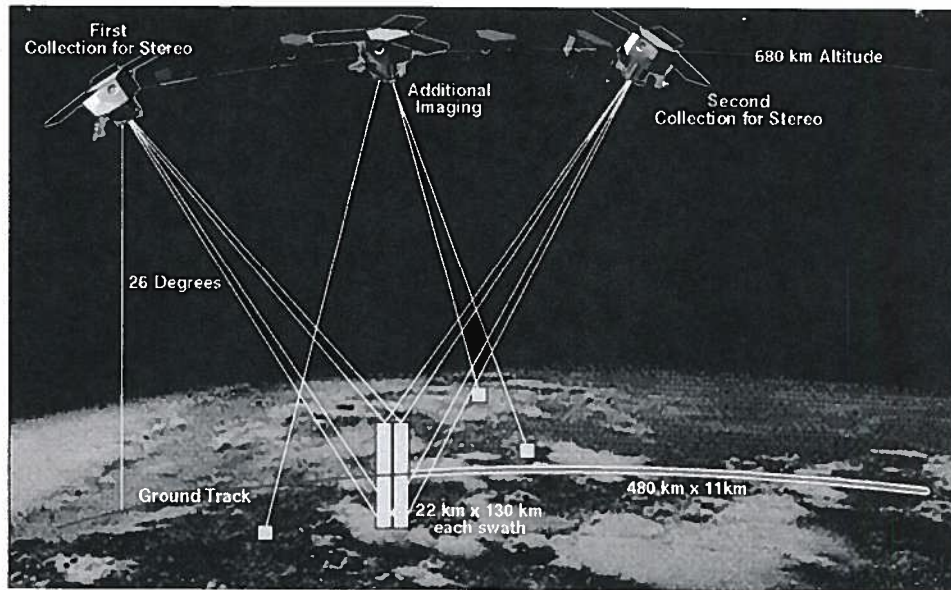


Figure 1 – Stéréo satellite image (Wikipedia)

Glossaire (1/2)

Les définitions fournies ici ont pour objectif d'éclairer la lecture de cette thèse.

Amélioration d'image :

Opération de traitement d'image qui améliore la visibilité des détails et caractéristiques d'une image. Généralement utilisée pour la perception humaine.

Apprentissage :

L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré. L'apprentissage neuronal fait appel à des exemples de comportement.

Arête :

Variation des valeurs de pixels supérieures à un seul. Les arêtes représentent les frontières entre les régions d'un objet ou d'une scène.

Artefact :

Structure créée artificiellement (accidentellement ou volontairement), se présentant par une forme quelconque faisant partie du fond de l'image et qui est utilisée pour assister les mesures effectuées ou pour localiser les objets.

Base d'apprentissage :

Exemples utilisés pour l'apprentissage, représentatifs du comportement de la fonction à réaliser.

Généralisation :

Principal intérêt des réseaux de neurones artificiels. Capacité à répondre correctement à des situations inconnues (c.a.d. n'apparaissant pas dans la base d'apprentissage).

Neurones :

Unité de traitement de l'information dans le cerveau, au nombre mille milliards environ.

Glossaire (2/2)

Processeur élémentaire :

Allusion au processeur de calcul (en beaucoup plus simple), aussi appelé neurone ou cellule.

Réseaux de neurones artificiels :

Réseaux fortement connectés de processeurs (neurones ou cellules) fonctionnant en parallèle. Chaque processeur élémentaire calcule une sortie unique sur la base des informations qu'il reçoit. Toute structure hiérarchique de réseaux est évidemment un réseau.

Rétro-action (rétropropagation) :

Technique d'entraînement ajustant les poids des couches d'entrée et cachées d'un réseau de neurones, pour la correction de ceux-ci à l'application d'un ensemble de vecteurs de données à l'entrée du réseau.

Supervisé :

Les poids sont modifiés en fonction de la sortie désirée.

Synapse :

Jonction entre deux neurones, très complexe au niveau des mécanismes chimiques mis en œuvre ; outrageusement simplifiée dans les modélisations.

Traitement d'image :

Manipulation numérique d'une image pour améliorer la visibilité de certaines caractéristiques, prendre des mesures ou en altérer le contenu.

Vision artificielle :

Utilisation de procédés avec l'acquisition optique sans contact pour recevoir et interpréter automatiquement l'image d'une scène réelle, dans le but d'obtenir de l'information et/ou le contrôle de machines ou de procédés.