



Université ABBES LAGHROUR Khenchela
Faculté des Sciences et de la Technologie
Département de Génie Industriel
جامعة عباس لغرور خنشلة
كلية العلوم والتكنولوجيا
قسم الهندسة الصناعية



N° Série :

Mémoire de fin d'étude

Pour l'obtention du diplôme de Master

Filière : Télécommunications

Spécialité : Télécommunications Avancées

THEME

**Compression des images satellites en
utilisant l'analyse en composantes
principales et la DCT**

Réalisé par : - OUNISSI Said

Dirigé par : Mr. DOUAK Fouzi

- ZITOUNA Hamed Bedreddine

Promotion 2015/2016

Je dédie ce travail à :

Ma mère,

Mon père,

Mes frères et mes sœurs,

Tous mes amis sans exception.

Ounissi said

Remerciements

Nous tenons à remercier premièrement Allah le tout puissant pour avoir donné le courage et la patience pour terminer ce travail.

*Nous remercies ensuite notre encadreur Monsieur **Fouzi Douak**, enseignant à l'université de Khenchela, département du Génie Industriel, pour avoir honoré par son encadrement, ses conseils précieux, sa patience et ses nobles valeurs humaines.*

Nos remerciements vont également aux membres de jury.

Enfin, nous exprimons notre remerciement à l'ensemble des personnes qui ont contribué à l'aboutissement de ce travail.

Résumé

L'objectif principal de ce travail, est l'étude de l'effet de la projection de l'analyse composante principale sur une image satellite multi-spectrale. Dans ce contexte, nous avons proposé une méthode de compression hybride basée sur la projection des données dans l'espace en composante principale (PCA), la transformée de la DCT, codage sans perte, et codeur Huffman.

Pour montrer l'efficacité de notre méthode proposée, nous testons plusieurs combinaison des paramètres (taille de bloc 8×8 et 16×16 , nombre de bits, seuillage), ceci est afin d'effectuer une étude comparative pour trouver la meilleur combinaison le plus adapté à la compression des images satellites.

Les résultats que nous avons obtenus, après simulation sont assez satisfaisants du point de vue amélioration de performances du codeur (PSNR, CR, bpp) par rapport à la norme JPEG.

L'application de l'approche proposée sur deux différentes images satellites (Khenchela et Lac Mono), a montré que : la meilleure performance peut être obtenue avec le changement d'espace originale vers un autre espace en composante principale, taille de bloc 16×16 , et nombre de bits égal a 9 sont les plus adaptées à cet algorithme.

Mots clés :

Compression des images multi-spectrales, DCT, IDCT, PCA, JPEG, Energie, Quantification, Huffman.

Abstract

The main objective of this work is the study of the effect of the using of projection of principal component analysis on a multispectral satellite image. In this context, we have proposed a hybrid compression method based on the projection data on a new space of principal component (PCA), the DCT transform, lossless coding, and Huffman coder.

To show the effectiveness of our proposed method, we test several combinations of parameters (block size 8×8 and 16×16 , the number of bits, thresholding), this is to conduct a comparative study to find the best combination more suited to the compression of satellite images.

The results obtained after the simulation are satisfactory from the standpoint of improvement encoder performance (PSNR, CR, bpp) compared to the standard JPEG.

The application of the proposed approach on two different satellite images (Khenchela and Mono Lake), showed that: the best performance can be obtained with the change of original space to another space in the component analysis, block size 16×16 , and the number of bits equal to 9 are most suited to this algorithm.

Key words:

compression image multi-spectrale image, DCT, IDCT , JPEG, PCA , energy, quantification , Huffman.

Liste des tableaux

| | |
|---|----|
| Tableau I. 1 Probabilités des symboles. | 12 |
| Tableau I. 2 Code de Huffman. | 14 |
| | |
| Tableau III. 1 Les performances mesurées (moyenne PSNR, moyenne CR) en appliquant l'approche sur les images (Khenchela, Lac Mono) reconstituées, les paramètres : deux type de découpage 8×8 et 16×16, varie le nombre de bits (7, 8 et 9bits), TH entre 5 et 50..... | 54 |

Liste des figures

| | |
|---|----|
| Figure I. 1 Elément d'une image numérique..... | 4 |
| Figure I. 2 Une image binaire..... | 7 |
| Figure I. 3 Image en niveaux de gris..... | 7 |
| Figure I. 4 Image en couleurs..... | 8 |
| Figure I. 5 Illustration d'une image multi-spectrale..... | 9 |
| Figure I. 6 Chaîne de traitement des images satellites..... | 10 |
| Figure I. 7 Génération de l'étiquette pour la séquence "acaab"..... | 13 |
| Figure I. 8 Arbre binaire de Huffman..... | 13 |
| Figure I. 9 Exemple de Codage par plage RLE..... | 14 |
| Figure I. 10 Codage par plage des coefficients AC dans JPEG..... | 15 |
| | |
| Figure II. 1 Image Multispectrale de la ville Khenchela..... | 22 |
| Figure II. 2 Image RVB de la ville Khenchela..... | 22 |
| Figure II. 3 Image Multispectrale de 7 bandes de Lac Mono..... | 23 |
| Figure II. 4 Image RVB du Lac Mono..... | 24 |
| Figure II. 5 L'énergie sur les quatre plans de l'image originale Khenchela..... | 25 |
| Figure II. 6 L'énergie sur les quatre plans de l'image de Khenchela après l'application de l'algorithme de PCA. PC P (P=1 :4): la projection des données en composantes principales de plan P de l'image satellite..... | 26 |
| Figure II. 7 L'énergie sur les quatre plans de l'image de Khenchela avant et après l'application de l'algorithme de PCA..... | 26 |
| Figure II. 8 L'énergie sur les sept plans de l'image de Lac Mono avant l'application de l'algorithme de PCA. Plan P (P=1 :7): espace originale de plan P de l'image satellite..... | 27 |
| Figure II. 9 L'énergie sur les sept plans de l'image de Lac Mono après l'application de l'algorithme de PCA. PC P (P=1 :7): la projection des données en composantes principales de plan P de l'image satellite..... | 28 |

| | |
|---|----|
| Figure II. 10 L'énergie sur les sept plans de l'image de la base de données Lac Mono avant et après l'application de l'algorithme de PCA..... | 28 |
| Figure III. 1 Principe de la compression de l norme JPEG. | 32 |
| Figure III. 2 Matrice de quantification pour la luminance (recommandation JPEG)..... | 34 |
| Figure III. 3 Matrice de quantification pour les chrominances (recommandation JPEG)..... | 35 |
| Figure III. 4 Schéma globale de la compression des images multispectrale. | 35 |
| Figure III. 5 Schéma globale de reconstruction des images satellites..... | 36 |
| Figure III. 6 Bloc du codage sans perte..... | 37 |
| Figure III. 7 Evaluation du taux de compression (<i>CR</i>) et nombre de bit par pixel (<i>bpp</i>) en fonction de <i>PSNR</i> entre la méthode proposée (DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de PCA sur l'image de la ville de Khenchela, pour les nombres de bits 7,8 et 9bits. | 40 |
| Figure III. 8 Evaluation du taux de compression (<i>CR</i>) et nombre de bit par pixel (<i>bpp</i>) en fonction de <i>PSNR</i> entre la méthode proposée (DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de PCA sur l'image de Lac Mono, pour les nombres de bits: 7, 8, 9 bits. | 41 |
| Figure III. 9 Evaluation du taux de compression (<i>CR</i>) et nombre de bit par pixel (<i>bpp</i>) en fonction de <i>PSNR</i> entre la méthode proposée (PCA+DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de codeur Huffman sur l'image de la ville de Khenchela, pour les nombres de bits: 7, 8, 9 bits. | 42 |
| Figure III. 10 Evaluation du taux de compression (<i>CR</i>) et nombre de bit par pixel (<i>bpp</i>) en fonction de <i>PSNR</i> entre la méthode proposée (PCA+DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de codeur Huffman sur l'image de Lac Mono, pour les nombres de bits: 7, 8, 9 bits..... | 43 |
| Figure III. 11 Evaluation du taux de compression (<i>CR</i>) et nombre de bit par pixel (<i>bpp</i>) en fonction de <i>PSNR</i> entre la méthode proposée (PCA+DCT+codage sans perte+Huffman, Bloc 16×16, nombre de bits: 9 bits), la norme JPEG, et PCA+JPEG, sur les deux images multispectrale : Khenchela et Lac Mono. | 44 |
| Figure III. 12 Compression d'une image de Khenchela (RVB) par l'approche proposée. | 47 |
| Figure III. 13 Compression d'une image de Khenchela (RVB) par l'algorithme PCA-JPEG..... | 48 |
| Figure III. 14 Compression d'une image de Lac Mono (RVB) par l'approche proposée..... | 51 |
| Figure III. 15 Compression d'une image de Lac Mono (RVB) par l'algorithme PCA-JPEG. | 52 |

Table des matières

| | |
|----------------------------|---|
| Introduction Générale..... | 1 |
|----------------------------|---|

Chapitre I

Généralités sur la compression d'image

| | |
|--|----|
| I.1. Introduction | 4 |
| I.2. Image numérique | 4 |
| I.3. Les caractéristiques d'une image numérique..... | 5 |
| I.3.1. Dimension..... | 5 |
| I.3.2. Résolution..... | 5 |
| I.3.3. Histogramme | 5 |
| I.3.4. Luminance | 5 |
| I.3.5. Contraste..... | 5 |
| I.3.6. Profondeur | 6 |
| I.3.7. Le poids de l'image..... | 6 |
| I.3.8. Contours et textures | 6 |
| I.4. Types d'image | 6 |
| I.4.1. Image binaire | 6 |
| I.4.2. Image d'intensité (niveau de gris) | 7 |
| I.4.3. Image couleur RVB | 8 |
| I.4.4. Images satellite multi-spectrales..... | 8 |
| I.5. Formats d'images | 10 |
| I.6. Codage sans perte | 11 |
| I.6.1. Codage arithmétique..... | 11 |
| I.6.2. Codage de Huffman..... | 13 |
| I.6.3. Le Codage par plage (Run length Encoding) | 14 |
| I.6.4. Codage différentiel | 15 |
| I.7. Codage avec perte..... | 15 |

| | |
|---------------------------|----|
| I.8. Quantification | 16 |
| I.9. Conclusion | 16 |

Chapitre II

Etude énergétique

| | |
|---|----|
| II.1. Introduction | 18 |
| II.2. Changement d'espace de couleur | 18 |
| II.3. Analyse en composantes principales | 19 |
| II.4. Etude énergétique | 20 |
| <u>II.4.1.</u> Image multispectrale kenchela | 20 |
| <u>II.4.2.</u> Image multispectrale Lac Mono | 22 |
| <u>II.4.3.</u> Résultats de l'étude énergétique | 24 |
| II.5. Conclusion | 29 |

Chapitre III

Compression des images satellites

| | |
|---|-----------|
| III.1. Introduction | 31 |
| III.2. Principaux algorithmes de compression | 31 |
| <u>III.2.1.</u> Standard JPEG | 31 |
| <u>III.2.1.1.</u> Transformée en cosinus discrète (DCT et IDCT) | 32 |
| <u>III.2.1.2</u> Calcul de la DCT et de l'IDCT | 33 |
| <u>III.2.1.3.</u> Quantification | 33 |
| <u>III.2.2.</u> Méthode proposée | 35 |
| <u>III.2.3.</u> Résultats sur les images satellites | 38 |
| III.3. Conclusion | 54 |
| Conclusion Générale | 57 |
| Bibliographie | 59 |

Introduction générale

Introduction générale

Le volume des images multi-spectrales acquises par satellites est actuellement en nette augmentation. Cela est principalement dû à l'évolution des besoins utilisateurs. Aujourd'hui ces derniers souhaitent des images disponibles dans un délai très court après leur acquisition et actualisées fréquemment. Ils ont besoin d'images multi-spectrales et d'une haute résolution. La compression des images satellites multi spectrale permet de réduire le volume des données transmises [1].

Dans la littérature scientifique Il existe plusieurs algorithmes de compression d'images, telles que : les algorithmes basées sur la DCT (Discrete Cosine Transform) [2], JPEG-DCT [3]. D'autres méthodes plus récentes, basées sur la transformée d'ondelette telles que : JPEG2000 [4], EZW (embedded coding using zerotrees of wavelet coefficients) [5] ou SPIHT (set partitioning inhierarchical trees) [6], etc., Le but principale de toutes ces algorithmes est d'obtenir un taux de compression élevé avec une qualité satisfaisante de l'image reconstruite. Généralement, les images contiennent trois plans de couleurs qui sont stockées sous le format RVB (Rouge Vert Bleu). Or le système RVB n'est pas le mieux adapté pour le processus compression, puisqu' il y a une forte corrélation entre les différents plans RVB; mais avec un changement d'espace de couleur d'image originale vers un autre espace moins corrélé, par exemple YCbCr, YUV, YIQ, O1O2O3 [7-11].

La transformation d'espace de couleurs permet une compression plus efficace, car l'information est principalement concentrée dans le premier plan. Nous pouvons, donc, comprimer plus efficacement les plans qui restent. Dans le cas des images a haute résolutions et les images satellites multispectrale, l'image contient plus de trois plans, donc on a bosien d'autres méthodes de changement d'espace qui supporte plus de trois plans. La projection en composante principale (PCA) [12], est une méthode similaire de ces méthodes précédentes, PCA permet de concentrée la plupart de la quantité d'énergie dans le premier plan, ce qui permet de compresser efficacement les images satellites. Donc en peut considérer que l'analyse en composantes principales est une méthode de prétraitement très importante pour la compression des images mutispectrale.

Nous distinguons deux types de compression des images : la compression sans perte, et avec perte [13]. La compression sans perte est complètement réversible, aucune perte d'information n'est introduite par les processus de compression/décompression, mais l'inconvenant majeure de cette catégorie que leur taux de compression et relativement faible. Par contre la compression avec perte permet des taux de compression très élevés avec une

dégradation plus importante. L'objectif des techniques proposées dans la littérature est de maximiser le taux de compression avec une qualité d'image acceptable.

Parmi les étapes les plus importantes dans la compression des images est la transformation. Dans notre travail, nous avons utilisé la transformée en cosinus discrète (DCT). Ensuite les coefficients obtenus après la DCT sont quantifiés. La phase de quantification introduit les erreurs dans le système de codage.

La méthode proposée présentée dans ce mémoire s'inspire de l'algorithme DCT-Adaptive Scanning [14], adaptée à la compression des images satellites. Après le découpage de l'image en blocs, un prétraitement d'espace originale vers un espace en composante principale, par l'intermédiaire d'une DCT. Ensuite une phase de quantification et seuillage, par la suite une étape de compression sans perte, à la fin une en applique le codeur Huffman.

L'organisation générale du mémoire est décrite ci-dessous :

Le chapitre I est consacré à la présentation des différents types d'images (binaires, niveaux de gris, couleur et les images satellites) et aussi les notions générales sur la compression des images avec perte et sans perte.

Le chapitre II une étude énergétique sur les images en utilisant la projection des données PCA.

Le chapitre III traite deux parties. Dans la première partie nous présentons la norme de compression JPEG, dans la deuxième partie, nous présentons une nouvelle méthode de compression des images satellites basé sur la projection des données dans un autre espace PCA, la transformée en cosinus discrète (DCT), la compression sans perte, et codage de Huffman, appliqué sur deux images satellite (Khenchela et Lac Mono).

La dernière partie de ce mémoire est consacrée à la conclusion générale.

Chapitre I

Généralités sur la compression des images

I.1. Introduction

L'utilisation de données sous leurs formes numériques ne serait pas possible aujourd'hui sans la compression préalable de celle-ci, et ceci pour plusieurs raisons :

- Les capacités de stockage des supports.
- La durée de transmission de ces données numériques est conditionnée par le débit du réseau qui est utilisé et qui est parfois relativement faible.

La compression des données numériques permet donc de diminuer la taille de stockage et de rendre possible leur transport sur les réseaux de communication (internet, GSM, câble, TV satellite).

Les supports sont digitaux et l'unité de mesure est le bit. Donc la compression d'image revient à réduire son espace en bits quelque soit le type d'image à compresser.

Vue l'intérêt de l'information à étudier ou à transmettre, on peut distinguer deux types de compression.

- La compression sans perte, là où il n'y a aucune perte de donnée de l'information originale.
- La compression avec perte, qui elle présente des dégradations par rapport à l'information originale.

Nous allons dans ce chapitre présenter quelques notions fondamentales de l'image numérique puis nous abordons les concepts de la compression des images satellites.

I.2. Image numérique

Une image numérique est stockée en mémoire sous forme de collection de points élémentaires appelés pixels. Nous pouvons considérer une image numérique comme une page de nombres organisés en matrice. Chaque nombre représente l'intensité du pixel. La position de chaque pixel peut être exprimée par deux coordonnées sur l'axe horizontal X et l'axe vertical Y comme le montre la figure ci-dessous.

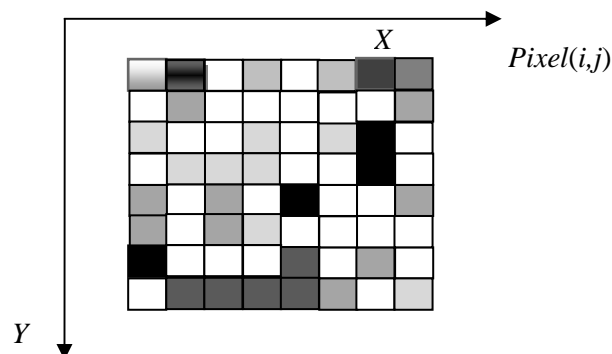


Figure I. 1 Élément d'une image numérique.

I.3. Les caractéristiques d'une image numérique

L'image est un ensemble structuré d'informations caractérisé par les paramètres suivants:

I.3.1. Dimension

C'est la taille de l'image, ou le nombre de points (pixels). Cela correspond au nombre de pixels qui compose l'image en hauteur (axe vertical) et en largeur (axe horizontal), par exemple, une image possédant 640 pixels en largeur et 480 en hauteur aura une définition de 640 pixels par 480, notée 640x480 [15].

I.3.2. Résolution

La résolution d'une image est définie par un nombre de pixels par unité de longueur classiquement en dpi (dots per inch) ou ppp (points par pouce) [16]. Ce paramètre est défini lors de la numérisation et dépend principalement des caractéristiques du matériel utilisé lors de la numérisation. Augmenter la résolution peut entraîner des temps de visualisation et d'impression plus longs, et conduire à une taille trop importante du fichier contenant l'image.

I.3.3. Histogramme

Un histogramme est un graphique statistique permettant de représenter la distribution des intensités des pixels d'une image, c'est-à-dire le nombre de pixels pour chaque intensité lumineuse.

L'objectif de l'histogramme est, diminuer les erreurs de quantification, comparer deux images obtenues sous des éclairages différents, ou encore pour mesurer certaines propriétés sur une image, on modifie souvent l'histogramme correspondant.

I.3.4. Luminance

C'est le degré de luminosité des points de l'image. Elle est définie aussi comme étant le quotient de l'intensité lumineuse d'une surface par l'aire apparente de cette surface, pour un observateur lointain, le mot luminance est substitué au mot brillance, qui correspond à l'éclat d'un objet [17].

I.3.5. Contraste

C'est l'opposition marquée entre deux régions d'une image, plus précisément entre les régions sombres et les régions claires de cette image. Le contraste est défini en fonction des luminances de deux zones d'images.

I.3.6. Profondeur

C'est le nombre de bits par pixel, cette valeur reflète le nombre de couleurs ou de niveaux de gris d'une image, par exemple :

- ✓ 32 bits/pixel = 1,07 milliards de couleurs.
- ✓ 24 bits = 16,7 millions de couleurs.
- ✓ 16 bits = 65 536 couleurs.
- ✓ 8 bits = 256 couleurs.

I.3.7. Le poids de l'image

Le poids d'une image se détermine en fonction de ces deux paramètres: dimensions, profondeur. Le poids de l'image est alors égal à sa dimension multipliée par sa profondeur.

Par exemple, pour une image 640x480 en vraies couleurs (True colors):

- ✓ Nombre de pixels (dimension) : $640 \times 480 = 307200$.
- ✓ Poids de chaque pixel (profondeur) : 24 bits = 3 octets.
- ✓ Le poids de l'image est ainsi égal à : $307200 \times 3 = 921600$ octets.

I.3.8. Contours et textures

Les contours représentent les frontières entre les objets de l'image, ou la limite entre deux pixels dont les niveaux de gris représentent une différence significative [17]. Les textures décrivent la structure de ceux-ci. L'extraction de contour consiste identifier dans l'image les points qui séparent deux textures différentes.

I.4. Types d'image

Une image numérique peut être vue comme une matrice. Chaque élément de cette matrice est appelé pixel. L'image peut être représentée de différentes manières au niveau informatique : image binaire, image d'intensité (niveau de gris), image couleurs contient trois plans (RVB), image multi-spectrale (plus de trois plans). Les images satellites sont de même nature et possèdent des propriétés similaires à la plupart des images photographiques, par exemple une image de la ville de Khenchela vu par Quickbird (voir Figures I. 2 à I.4).

I.4.1. Image binaire

C'est une matrice où ses éléments ne peuvent prendre que deux valeurs 0 et 1. Les zéros représentent le noir et les uns représentent le blanc.

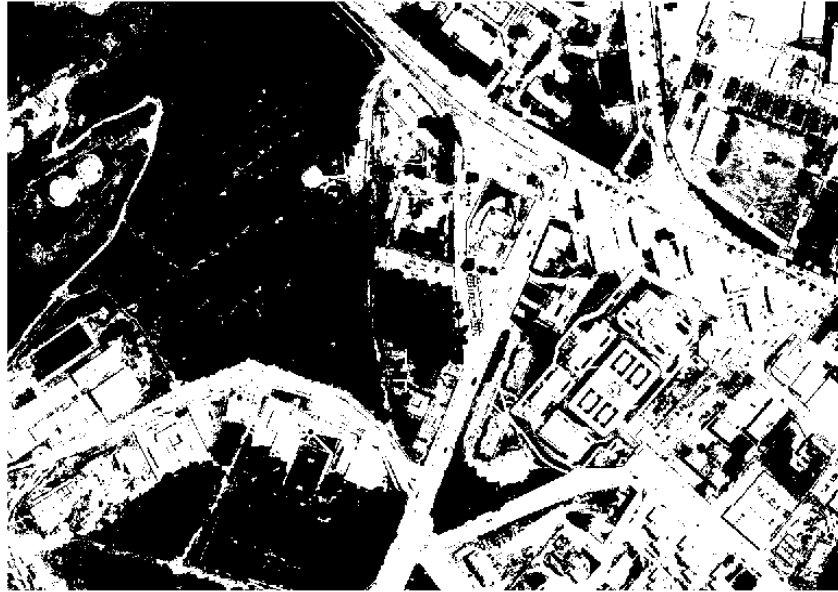


Figure I. 2 Une image binaire.

I.4.2. Image d'intensité (niveau de gris)

Une image d'intensité est une matrice dans laquelle chaque élément est un nombre réel compris entre 0 et 255. On parle aussi d'image en niveau de gris.



Figure I. 3 Image en niveaux de gris.

I.4.3. Image couleur RVB

L'image couleur est formée par les trois couleurs de base, le rouge, le vert et le bleu. Pour présenter la couleur d'un pixel il faut donc donner 3 nombres. Qui correspondent à l'intensité des trois couleurs de base.



Figure I. 4 Image en couleurs.

I.4.4. Images satellite multi-spectrales

Les images acquises par satellite offrent une perspective unique de la terre. La caractéristique principale des images traitées par un satellite est leur grande taille.

Une image multi-spectrale est une image qui regroupe plusieurs plans d'image appelés bandes spectrales, ayant un nombre de D bandes est illustrée dans la Figure I.5. Les composantes du vecteur X sont les valeurs du niveau de gris des pixels qui se trouvent dans la même position dans les différentes bandes. La matrice X est de dimension D , et le nombre de ces vecteurs est égal à la résolution de la bande.

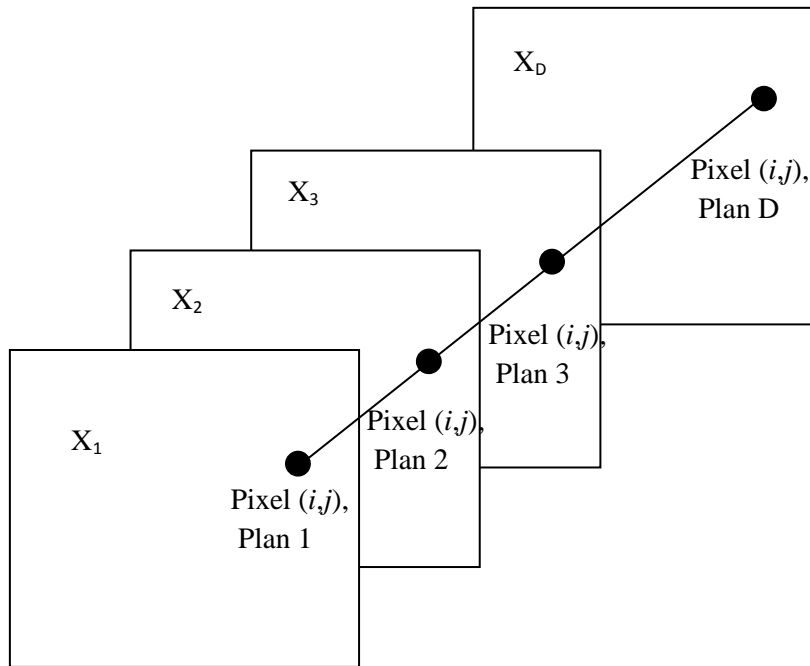


Figure I. 5 Illustration d'une image multi-spectrale.

La chaîne de traitement des images satellite est représentée sur la Figure I.6. Un satellite acquiert les images par balayage en défilant autour de la Terre. Les informations géométriques sur le satellite (orbite, altitude, la direction de visée, localiser la zone d'image) sont enregistrées au cours de chaque acquisition. Les informations radiométriques seront transmises au sol afin d'effectuer une première série de corrections sur l'image brute. Les images sont compressées puis, soit directement transmises au sol lorsque le satellite est en visibilité directe avec une station de réception. Au sol, plusieurs étapes de traitements sont effectuées. Parmi ces étapes est la décompression des images, on obtient donc une image brute. Ensuite plusieurs types de corrections et des traitements sont effectués sur l'image brute (corrections radiométriques, traitements géométriques). Cette image peut déjà être commercialisée. Les images satellites sont très riches en informations (géographiques ou cartographiques).

Il existe plusieurs types de satellites artificiels pour l'objectif de capter l'imagerie satellitaire, parmi ces satellites : Le satellite QuickBird qui offre la combinaison d'une résolution spatiale très élevée, et d'un haut degré de précision pour le géoréférencement. Il est aussi capable d'acquérir annuellement des données de très grande couverture. QuickBird collectait des images multispectrales à 2,4 m de résolution et des images panchromatiques à 0,6 m de résolution. La compression est une étape très importante dans la chaîne de traitement des

images satellites vu la plus grande quantité d'information, dans ce mémoire, nous proposons de nouvelle technique pour l'objective de la compression des images satellites.

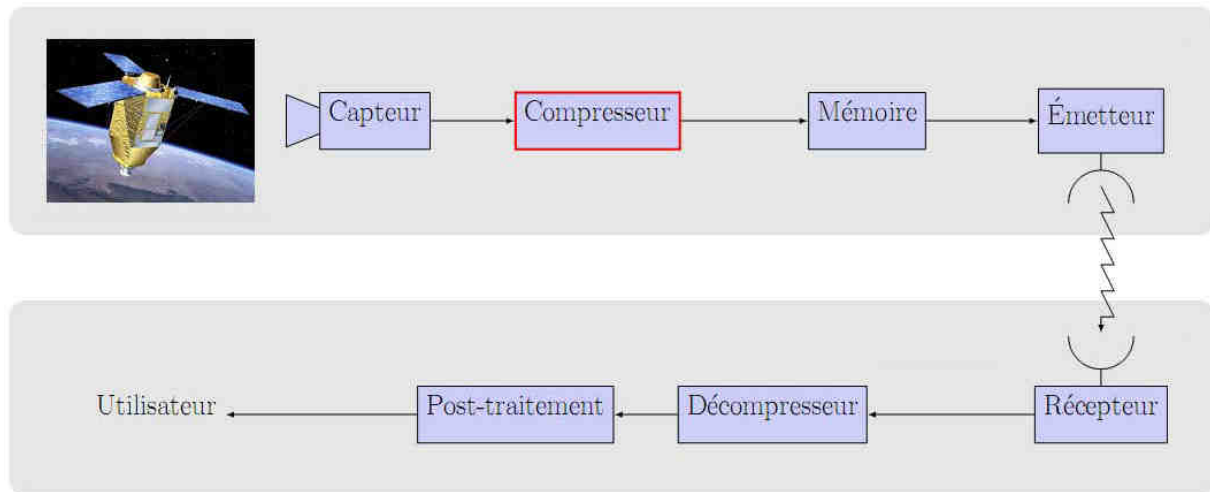


Figure I. 6 Chaîne de traitement des images satellites.

I.5. Formats d'images

Il existe plus d'une cinquantaine de types de formats d'image. Pour chacun d'entre eux la structuration des données et les attributs sont différents. La standardisation d'un format d'image permet de régler l'utilisation, la divulgation et la production de logiciels et de hardware compatibles avec le format standard.

Le plus populaire de ces standards est le JPEG, il a été créé vers la fin des années 80. JPEG utilise les principaux modes suivants : baseline, lossless, progressive et hiérarchique. Le mode baseline est le plus utilisé, il supporte le codage avec perte seulement. Le mode lossless est moins populaire il ne supporte pas le mode avec perte.

JPEG2000 est le dernier standard ISO/ITU-T pour le codage des images fixes, est basé sur la DWT, la quantification scalaire, la modélisation du contexte, le codage arithmétique et l'allocation débit post-compression.

Le JPEG2000 possède des fonctionnalités supplémentaires par rapport au format JPEG. La résistance aux erreurs est une caractéristique particulière du JPEG2000. Après le codage entropique plusieurs caractères de contrôle (segment marks, resynchronising marks) sont insérés dans le flux de bits. Cette démarche est faite pour synchroniser les informations, limiter la taille du segment et éviter la propagation des erreurs.

Une autre fonctionnalité importante du JPEG2000 est la compression par région d'intérêt (ROI). Ceci permet d'avoir des taux de compression différents dans certaines régions de

l'image. Les zones importantes peuvent être compressées quasi sans pertes et les zones moins importantes avec un fort taux de compression.

Malgré ses nombreuses fonctionnalités, le JPEG2000 possède quelques inconvénients. Il nécessite entre deux et six fois plus de cycles de CPU que JPEG et il n'est pas indiqué pour les machines avec faibles ressources, la plupart des appareils numériques (appareils photos, caméscopes, téléphones portable, etc.) et les logiciels qui capturent et traitent les images sont au format JPEG. L'algorithme JPEG est beaucoup moins complexe et il peut être implémenté en hardware [18, 19].

I.6. Codage sans perte

La compression sans perte (codage réversible) permet de retrouver la valeur exacte du signal ou image comprimé lorsqu'il n'y aucune perte de données sur l'information d'origine. En fait, la même information est réécrite d'une manière plus concise. Le processus de codage sans perte crée des "mots-codes" à partir d'un dictionnaire statique ou d'un dictionnaire construit dynamiquement. Ces processus s'appuient sur des informations statistiques de l'image. Les codes statistiques les plus répandus sont le codage de Huffman et le codage arithmétique. Le codage statistique permet de s'approcher au mieux de l'entropie. Ils ont pour principe d'associer aux valeurs les plus probables les mots binaires les plus courts [13].

I.6.1. Codage arithmétique

Le codage arithmétique se singularise par sa capacité à coder chaque symbole sur un nombre non entier de bits. En réalité, il n'assigne pas un mot de code à chaque symbole mais il associe un point de l'intervalle $[0,1]$ à un ensemble de symboles. Le principe repose sur le découpage de l'intervalle $[0,1]$. Chaque symbole se voit attribuer une partition de l'intervalle dont la taille est égale à sa probabilité d'occurrence. L'ordre de rangement est mémorisé pour être utilisé lors du décodage.

Le codage arithmétique est généralement plus performant que le codeur de Huffman. Il tend vers la limite inférieure théorique mais cependant il est gourmand en ressources et nécessite de connaître à priori l'intégralité du signal avant de pouvoir procéder au codage

Les différentes étapes de l'algorithme de codage sont :

- L'initialisation : nous affectons à chaque symbole une plage d'intervalle basée sur sa probabilité d'apparition fournie par le modèle. Les bornes externes de cet intervalle sont zéro et un.

- Le traitement du message : nous initialisons un intervalle de travail en prenant comme bornes zéro et un. Le premier symbole est représenté par la plage qui lui est affectée à l'étape 1. Chaque symbole suivant restreint davantage l'intervalle et il est représenté par sa plage relative dans la plage précédente. Ainsi le flot de données est traduit par un nombre contenu dans la dernière plage calculée.
- On rajoute un symbole spécial pour déterminer la fin du message où l'on donne la longueur du flot avec le message codé pour permettre au décodeur de déterminer la fin du message.

Par exemple, supposons qu'on veut coder une partie "acaab" d'une longue séquence avec une probabilité d'apparition indiquée dans le Tableau I.

| Symbole | Probabilité | Intervalle |
|----------|-------------|------------|
| <i>A</i> | 0.7 | [0, 0.7] |
| <i>B</i> | 0.1 | [0.7, 0.8] |
| <i>C</i> | 0.2 | [0.8, 1.0] |

Tableau I. 1 Probabilités des symboles.

L'intervalle initial [0, 1] va être divisé en trois sous-intervalles suivant les probabilités des symboles de la séquence. Ce qui donne les sous-intervalles suivants: [0, 0.7], [0.7, 0.8] et [0.8, 1.0].

Dans cette exemple, le 1er symbole est "a", l'étiquette appartient donc à l'intervalle [0, 0.7]. Après que le 1er symbole soit codé, les limites inférieures et supérieures de l'intervalle sont respectivement 0 et 0.7 pour le symbole suivant. L'intervalle [0, 0.7] va être divisé en trois sous-intervalles : [0, 0.49], [0.49, 0.56] et [0.56, 0.7] correspondant respectivement aux symboles "a", "b" et "c".

Le 2ème symbole est "c", de probabilité 0.2. Ainsi, le nouveau sous-intervalle sera [0.56, 0.7]. Ce dernier va être divisé en trois sous-intervalles : [0.56, 0.658], [0.658, 0.672] et [0.672, 0.7].

Le 3ème symbole est "a". Le nouveau sous-intervalle sera en conséquence [0.56, 0.658]. Le sous-intervalle [0.56, 0.658] va être partagé à son tour en trois sous-intervalles : [0.56, 0.6286], [0.6286, 0.6286] et [0.6286, 0.658].

Le 4ème symbole est "a", donc le nouvel intervalle est [0.56, 0.6286].

Et ainsi de suite jusqu'à que l'on code la totalité de la séquence. On aura à la fin, le sous-intervalle suivant [0.60802, 0.61488]. Un nombre contenu dans l'intervalle final comme

0.60803 code sans ambiguïté le message “acaab”. La Figure I.7 est une représentation de ce processus.

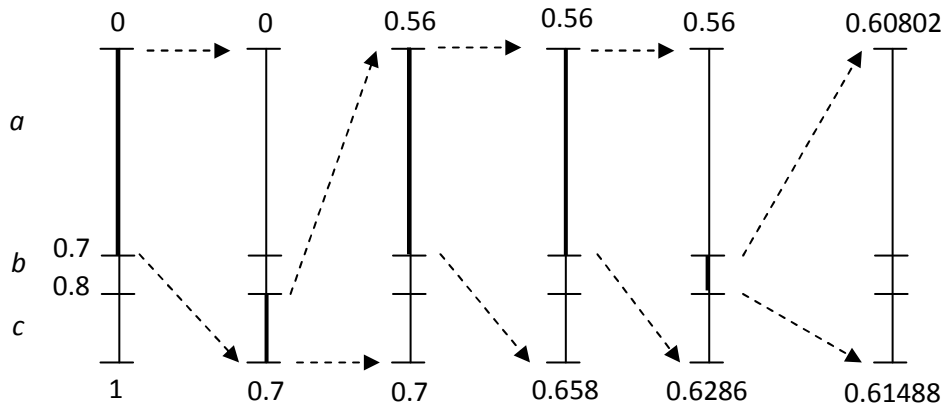


Figure I. 7 Génération de l’étiquette pour la séquence “acaab”.

I.6.2. Codage de Huffman

Le codage de Huffman consiste à coder les symboles par une représentation de bits à longueur variable. Les symboles ayant la probabilité d’apparition forte sont codés avec des chaînes de bits plus courtes, tandis que les symboles dont la probabilité d’apparition est faible sont codés par des chaînes plus longues. Le code d’un symbole ne doit pas être le préfixe d’un autre code. Cette propriété est admise afin que la reconnaissance soit possible. Pour représenter le codage de Huffman, on utilise l’arbre binaire [20].

Soit un message a codé “ABBBBAAC”. La fréquence d’apparition ainsi que le code Huffman correspondant est donné dans le Tableau I.2 et représentés par la Figure I.8.

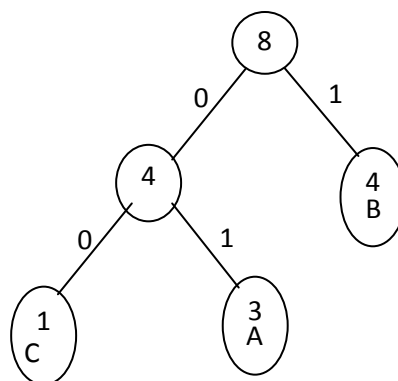


Figure I. 8 Arbre binaire de Huffman.

| Symbole | A | B | C |
|------------------------|----|---|----|
| Fréquence d'apparition | 3 | 4 | 1 |
| Code Huffman | 01 | 1 | 00 |

Tableau I. 2 Code de Huffman.

I.6.3. Le Codage par plage (Run length Encoding)

Le procédé Run-Length ne relève pas d'une théorie mathématique très complexe. Il s'agit simplement de remplacer des éléments significatifs, successifs et identiques par un seul d'entre eux, suivi du nombre de répétitions [21]. Un exemple de traitement RLE est donné à la Figure I.9 Ce procédé peut paraître simpliste et peu performant si on cherche à l'appliquer, par exemple, à un texte, les répétitions de lettres n'apporteraient qu'une compression dérisoire! En revanche, si on l'applique à une image, il est aisé de s'apercevoir que les plages de couleurs (ou niveaux de gris) homogènes sont souvent importantes, même si leur nombre est parfois faible.

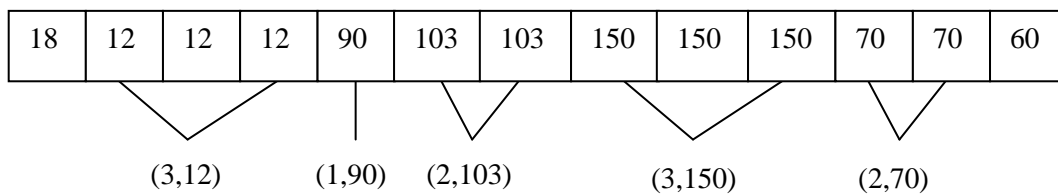


Figure I. 9 Exemple de Codage par plage RLE.

Si n octets successifs sont dans un même état, il est aisé de transmettre l'octet répété et le nombre de répétitions. On pourra ainsi, dans la plupart des cas, coder sur 3 octets les n octets composant le signal initial. S'il est relativement simple de coder l'octet à répéter, suivi du nombre de répétitions dans l'octet suivant, cette méthode peut se révéler très pénalisante pour certains cas : à la limite si deux octets consécutifs sont toujours différents, la taille des données compressées sera la double de celle des données initiales. Pour éviter cet inconvénient, les versions les plus avancées du RLE utilisent un code discriminant pour indiquer le début d'une séquence octet à répéter + le nombre de répétitions, les octets isolés restant codés sous leur forme initiale

Dans la norme JPEG (Joint Photographic Experts Group) le "Codage par plage" se focalise sur les répétitions des "zéros" dans une séquence (ou un balayage). Ainsi les coefficients AC quantifiés dans un balayage subissent un codage par plage avant le passage par le codeur de

Huffman. La Figure I.10 représente l'implantation du "Codage par plage" des coefficients AC (les 63 coefficients de fréquence non nulle pour un bloc 8×8) dans la norme JPEG.

| | | | | | | | | | | | | | |
|-----|---|---|-----|---|---|---|---|----|----|----|-----|----|----|
| 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | ... | 63 | 64 |
| 412 | 0 | 0 | -23 | 9 | 0 | 0 | 0 | 0 | 0 | 6 | ... | 0 | 0 |

La première ligne indique l'ordre du coefficient AC dans le balayage en zig-zag. La deuxième ligne indique la valeur du coefficient AC quantifié.

(0,412), (2,-23), (0,9), (5,6), ..., EOB

Figure I. 10 Codage par plage des coefficients AC dans JPEG.

I.6.4. Codage différentiel

La première étape de codage de la matrice quantifiée est le codage différentiel pour le coefficient DC (la composante continue du bloc).

La modulation différentielle est basée sur le constat que les données analogiques ont tendance à varier de valeur d'une manière progressive, les sauts importants du signal étant exceptionnels. En effet, les blocs adjacents sont très fortement corrélés. Cette opération transforme donc les valeurs absolues en valeurs relatives. Pour les données graphiques, elle permet de transmettre non plus la valeur moyenne d'un bloc de pixels mais la différence de valeur avec le bloc précédent. Par exemple si la valeur moyenne est codée sur huit bits, un système de codage différentiel peut coder la différence sur quatre bits, ce qui permet de réduire le nombre de bits nécessaires au codage. Pour la compression des images fixes, le problème vient du fait que les valeurs des pixels peuvent varier de manière importante. Le codage différentiel doit donc s'adapter aux petites et aux grandes différences entre les pixels, ce qui limite son efficacité. Par exemple certaines images montrent de longues étendues de pixels de même valeur, elles seront bien compressées, tandis que d'autres, présentant des variations de couleur importantes, seront peu ou pas compressées [22].

I.7. Codage avec perte

Les méthodes avec pertes ou irréversibles sont des méthodes qui tirent une partie d'une corrélation (ou redondance) existante dans l'image. L'information perdue est due à l'élimination de cette redondance, ceci rend possible une compression plus importante [23].

Pour améliorer le taux de compression, il va falloir perdre quelque chose c'est-à-dire dégrader l'image. Ceci permet d'atteindre des taux arbitrairement grand au prix d'une dégradation toujours plus importante. L'objectif des algorithmes de compression avec perte est de minimiser cette dégradation pour un taux de compression donné [24].

Dans un premier temps, afin de mieux compacter l'information, la source est transformée en groupe des coefficients. Les transformations les plus utilisées, que ce soit pour les images fixes ou les séquences d'images, sont la Transformée en Cosinus Discrète (DCT), la Transformée en Ondelettes Discrète (DWT) ou la décomposition Pyramidale.

Dans un second temps, les coefficients obtenus après la transformation sont quantifiés. La phase de quantification introduit l'erreur dans le système de codage [24], ce dernier est un des mécanismes utilisés dans les algorithmes de compression, qui produit des pertes d'information.

I.8. Quantification

Par définition, l'opération de quantification permet une transcription des données depuis un espace de taille "infini" constitué par exemple de l'ensemble des nombres flottants, ou de taille très importante, vers un espace contenant un nombre limité de coefficients [25]. La quantification fait partie de plusieurs méthodes de compression d'image. L'objectif est de réduire la taille des coefficients de façon que cette réduction n'apporte pas de dégradations visuelles à l'image [23]. Plusieurs méthodes de quantification ont été proposées dans la littérature scientifique, parmi ces méthodes les plus populaires : la quantification scalaire et la quantification vectorielle

I.9. Conclusion

Ce chapitre, nous avons présenté plusieurs techniques de compression d'images sans perte et avec pertes. Toutes les approches énumérées précédemment utilisent d'une manière ou d'une autre les corrélations entre pixels voisins dans l'image. Pour codage sans perte nous avons opté dans notre travail pour le codeur Huffman pour effectuer la compression des images ou en utilisant la méthode proposé dans le chapitre III.

Dans le chapitre suivant, nous allons présenter une étude énergétique sur les images multispéctrale.

Chapitre II

Etude énergétique

II.1. Introduction

Habituellement, les images couleur sont stockées sous le format RVB (Rouge Vert Bleu). Or le système RVB n'est pas le mieux adapté pour appliquer le codage JPEG. En effet, l'œil humain est plus sensible à la luminance qu'à la chrominance d'une image. Aussi une transformation est réalisée pour passer à un espace couleur, plus adapté à la compression. Dans la littérature plusieurs types de changement d'espace de couleur ont été proposés, le changement YCbCr, HSI, CMY, ... etc. Dans notre cas pour les images multispéctrale, nous avons choisi la projection des données dans l'espace de composante principale. Par la suite on présentera une étude énergétique de chaque plan de l'image multispéctrale afin de confirmer que l'espace PCA est meilleur espace adapté à la compression des images.

II.2. Changement d'espace de couleur

Toute longueur d'onde visible peut être visuellement simulée en convoluant le signal avec les fonctions de sensibilité des trois différents capteurs rétiniens du système visuel humain dit *LMS* (Large=565nm dit rouge, Medium=535nm dit vert, Short=430nm dit bleu). Dans le cas d'une compression avec perte, la reconstruction de chaque bande (*RVB*) risque de ne pas appréhender les structures de l'image de la même façon, engendrant différentes erreurs de reconstruction et par la même, de fausses couleurs visuellement choquantes. On préférera donc un espace de luminance et chrominance rouge et bleu *YCbCr* (ou *YUV*) où les primaires sont décorréliées, ce qui offre l'avantage de séparer les informations d'intensité lumineuse et de couleur. Un tel espace permet de gérer les premières avec plus de soin [26, 27]. Passage de l'espace *RGB* à l'espace *YCbCr* [13] est définie par l'équation suivante :

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.144 \\ -0.16875 & -0.33126 & -0.5 \\ 0.5 & -0.41869 & 0.08131 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (\text{II.1})$$

Les équations de passage des composantes *YCbCr* aux composantes *RGB* s'obtiennent facilement par l'équation ci-dessous :

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.402 \\ 1 & -0.34413 & -0.71414 \\ 1 & 1.772 & 0 \end{bmatrix} \times \begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} \quad (\text{II.2})$$

La transformation des composantes *RVB* en composantes *YCbCr* est une transformation bijective, il n'y a pas de pertes de données pendant cette étape.

Il existe d'autres modèles comme les espaces : *HSI* (Hue Saturation Intensity) et *HSV* (Hue Saturation Value), qui prennent en relation les notions intuitives de teinte, saturation, et luminance.

CMY(K) (Cyan Magenta Yellow (blacK)) utilisé pour les impressions couleur. *YIQ*, *YUV*, *YCbCr* (Luminance-Chrominance) utilisé en imagerie et en vidéo. Il existe également des espaces colorimétriques utilisés pour le traitement d'images comme les systèmes $I_1I_2I_3$, *XYZ*, *Luv* et *Lab* qui ont été définis par la *CIE* (Commission Internationale de l'Éclairage).

Dans notre cas les images multispectrales sont généralement de nombre de plans est supérieur à trois plans, pour cela en utilise l'analyse en composante principale.

II.3. Analyse en composantes principales

L'analyse en composantes principales, est une technique dont le but est d'expliquer la structure covariante d'une images mutispectrale dans un espace de plus petite dimension dont les nouvelles variables sont indépendantes (non corrélées) [28, 29]. On détermine les vecteurs propres de la matrice carrée R résultant de la multiplication de la matrice d'une image X (convertir la matrice de l'image multispectrale vers un vecteur de dimension $N \times M$) par sa transposée les vecteurs propres sont appelés composantes principales, et sont orthogonaux entre eux, leur covariance étant nulle. Géométriquement, cette transformation équivaut à une rotation des axes dans l'espace des m dimensions (variables d'origine est le nombre de plans de l'image multispectrale) dans un nouvel espace à p dimensions dont les axes décrivent la variance maximum des données. L'origine des axes se trouve alors au centre du nuage de points dans l'espace des mesures. Le premier axe principal sera celui qui rendra compte de la variance la plus grande (les valeurs propres étant classées par ordre de valeurs décroissantes). La variance de la nouvelle image de chaque vecteur propre sera égale à 1, soit par exemple pour le premier vecteur propre :

$$u_{11}^2 + u_{12}^2 + u_{13}^2 + \dots + u_{1m}^2 = 1 \quad (\text{II.3})$$

Pour obtenir les coordonnées des données le nouvel espace ('principal components scores' y_y), on projette chaque vecteur individu de l'image multispectrale (x_1, x_2, \dots, x_m) de X sur chaque composante principale en le multipliant par les éléments (u_j) de chaque vecteur propre U (composante principale).

La somme des valeurs propres était égale à la trace de la matrice des variances-covariances. Les axes principaux contiennent de plus l'ensemble de la variance de X (c'est le but de PCA).

On va pouvoir retrouver la contribution relative de chaque composante principale simplement en divisant sa valeur propre correspondante par la trace de variance totale.

Bien que pour m variables d'origine, m composantes principales (CP) soient calculées à fin de PCA, il est souvent possible de ne retenir que les deux premières composantes principale qui à elles seules recèlent la majeure partie de l'information des images multispectrale.

II.4. Etude énergétique

Pour justifier l'application de PCA, on mesure l'énergie dans chaque plan de l'image originale et l'image après l'application de PCA, premièrement on calcule l'énergie totale (ET) de l'espace originale par l'expression suivante :

$$ET = \sum_{k=1}^P \sum_{i=1}^N \sum_{j=1}^M x_k(i, j)^2 \quad (\text{II.4})$$

Où N , M est la dimension de l'image, et P nombre de plans de l'image multispectrale.

Ensuite on calcule l'énergie de chaque plan (E_k) par la relation suivante :

$$E_k = 100 \times \frac{ET_k}{ET} \quad (\text{II.5})$$

Cette mesure est réalisée sur deux images multispectrale. La première image satellite est l'image de la ville de Khenchela et le deuxième est l'image de Lac Mono (voir Figures II.1 à II.4).

II.4.1. Image multispectrale khenchela

La base de donnée de khenchela est une image satellitaire Multispectrale géoréférencée du radar Quickbird de 0.6 m/p en quatre hyperplan: (Blue: 430-545 nm, Green: 466-620 nm, Red: 590-710 nm, et Near-IR: 715-918 nm)

L'image multispectrale de Khenchela a dimension de [816×1154 pixels], chaque pixel est caractérisé par 4 plans, ce qui donne un total de taille réelle [816×1154×4] (Figure II.1). La version RVB correspondantes (3 première plans) est présentée dans la Figure II.2.

Plan 1



Plan 2



Plan 3





Figure II. 1 Image Multispectrale de la ville Khenchela.



Figure II. 2 Image RVB de la ville Khenchela.

II.4.2. Image multispectrale Lac Mono

Le lac Mono (Mono lake en anglais), a dimension de [512×512 pixels] de 7 plans (Figure II.3), est un lac salé situé dans le désert de la Sierra Nevada en Californie, à environ 13 km à

l'est du parc du Yosemite, près de la ville de Lee Vining. Figure II.4 représente l'image de Lac Mono dans l'espace RVB.

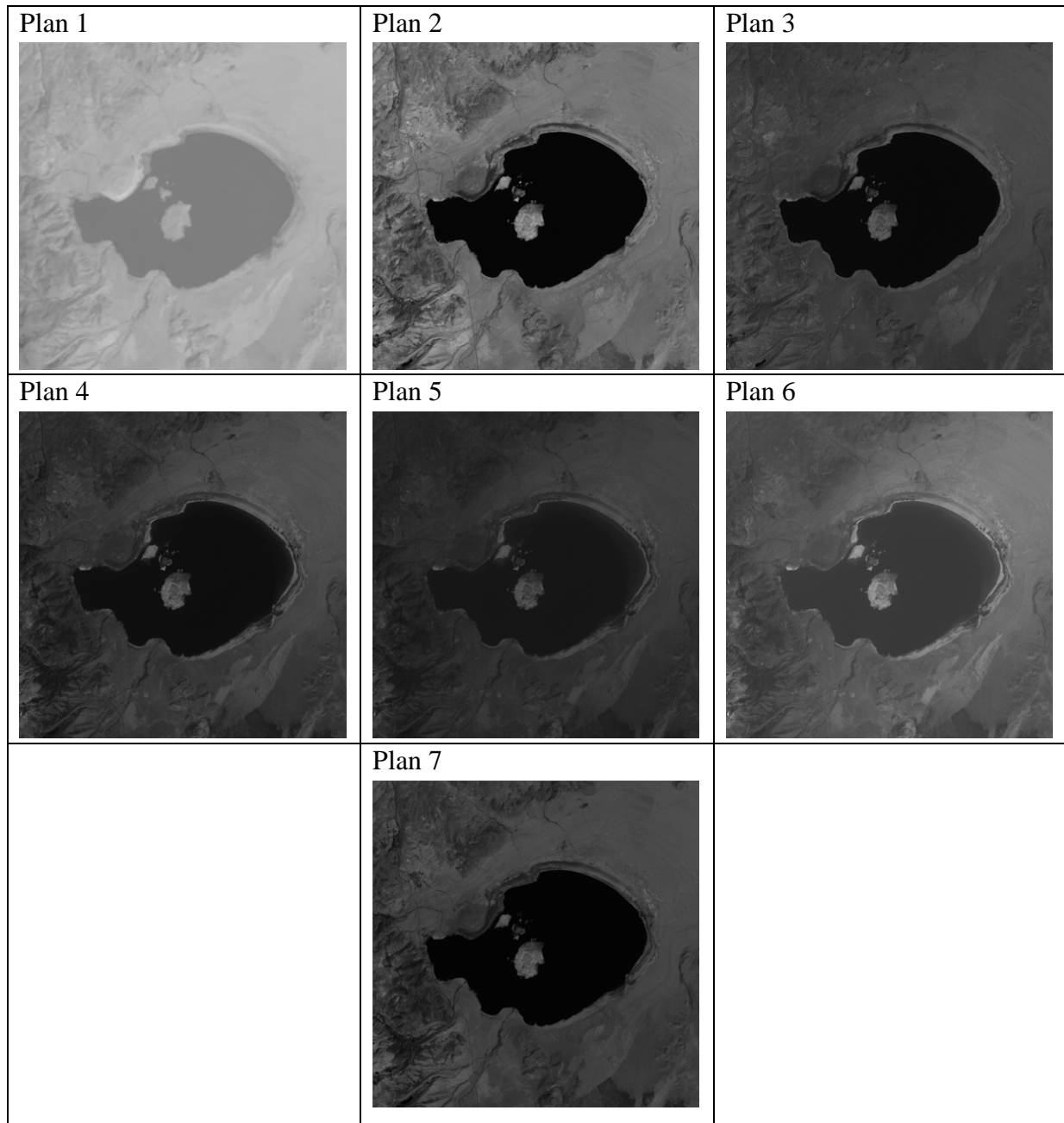


Figure II. 3 Image Multispectrale de 7 bandes de Lac Mono.



Figure II. 4 Image RVB du Lac Mono.

II.4.3. Résultats de l'étude énergétique

Représentation de concentration d'énergie sur chaque plan d'image multi spectrale originale (Figures II.5 et II.8) et après l'application de (PCA) sur chaque plan d'image originale, pour les deux images satellites Khenchela et Lac Mono (Figures II.6, et II.9). Les courbes obtenues avant et après l'application de l'algorithme de l'analyse en composante principale (PCA) sont représentée dans les Figures II.7, et II.10.

Les points de la courbe avec une couleur noire montrant la quantité d'énergie concentrée dans les quatre plans (Khenchela) et les sept plans (Lac Mono) qui ressemblent également rapprochement entre eux (degré élevé de corrélation entre les différents plans de l'espace originale), tandis que la courbe avec la couleur rouge après l'application de l'algorithme (PCA) indique que la plupart de la quantité d'énergie concentrée dans la première plan.

Application de la norme de l'analyse en composante principale sur les images satellites pour les deux bases de données de Khenkhela et Lac Mono permet de concentrer la majorité d'énergie dans les deux premiers composante principale, c-à-d, elle contient les informations principales de l'image, malgré la petite énergie qui reste qui on peut négliger mais on na besoin de cette partie pour la reconstruction des images compressées (application des étapes inverse de la chaine de compression).

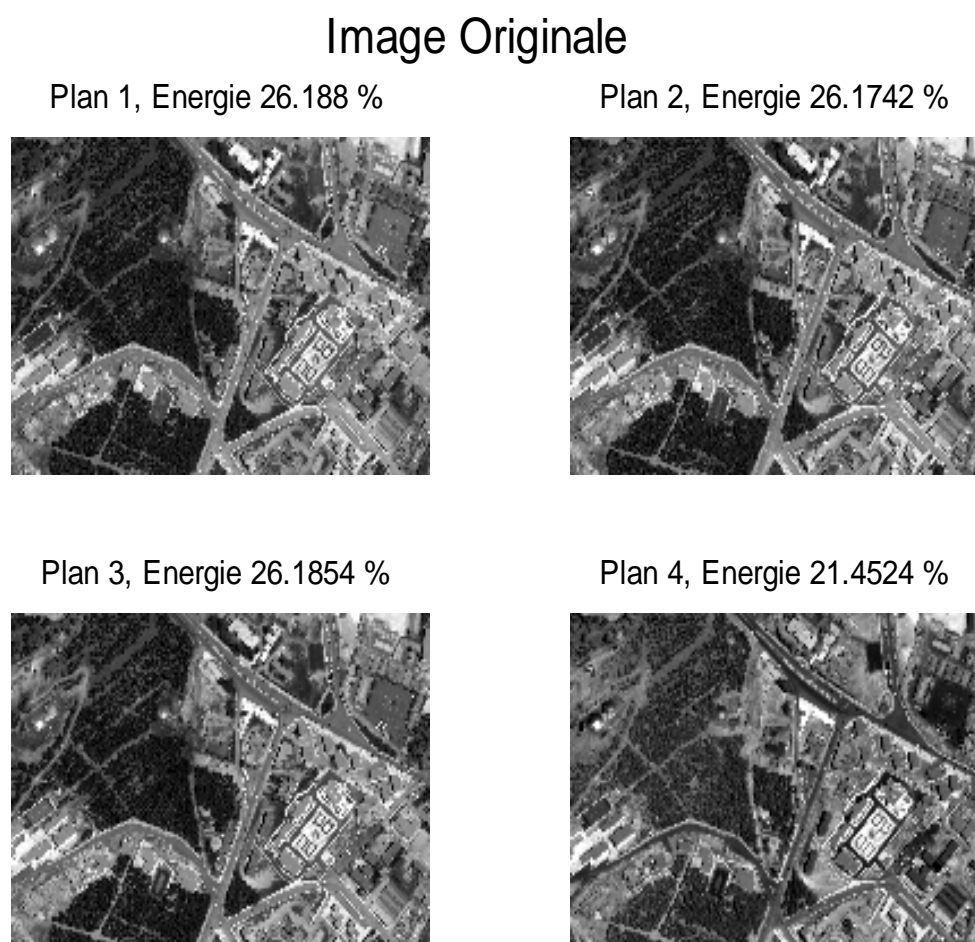


Figure II. 5 L'énergie sur les quatre plans de l'image originale Khenchela.

Analyse en composantes principales (PCA)

PC 1, Energie 92.9598 %



PC 2, Energie 6.5266 %



PC 3, Energie 0.46252 %



PC 4, Energie 0.051096 %



Figure II. 6 L'énergie sur les quatre plans de l'image de Khenchela après l'application de l'algorithme de PCA. PC P (P=1 :4): la projection des données en composantes principales de plan P de l'image satellite.

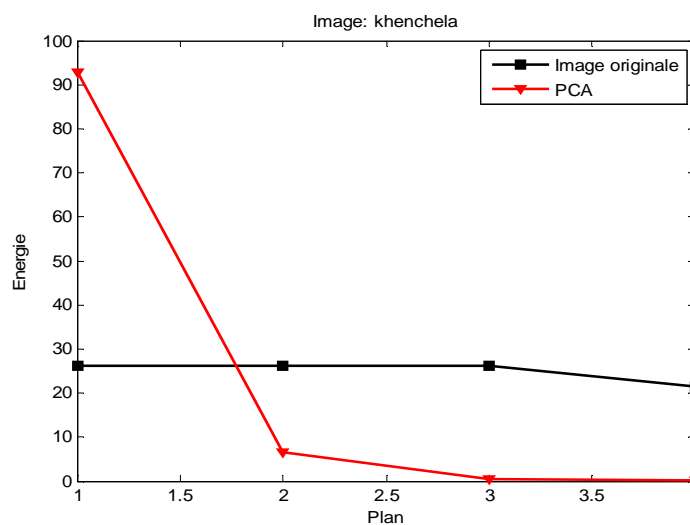
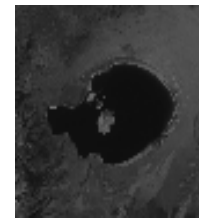
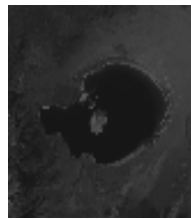
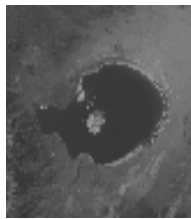


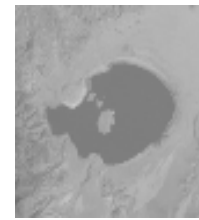
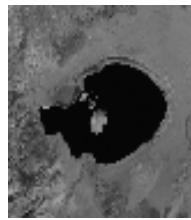
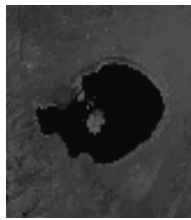
Figure II. 7 L'énergie sur les quatre plans de l'image de Khenchela avant et après l'application de l'algorithme de PCA.

Image Originale

Plan 1, Energie 10.9246 % Plan 2, Energie 4.4303 % Plan 3, Energie 10.2373 %



Plan 4, Energie 11.8496 % Plan 5, Energie 40.3132 % Plan 6, Energie 8.1015 %



Plan 7, Energie 14.1435 %

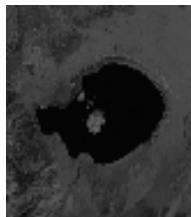


Figure II. 8 L'énergie sur les sept plans de l'image de Lac Mono avant l'application de l'algorithme de PCA. Plan P (P=1 :7): espace originale de plan P de l'image satellite.

Analyse en composantes principales (PCA)

PC 1, Energie 90.7098 % PC 2, Energie 6.6987 % PC 3, Energie 1.4326 %



PC 4, Energie 0.95192 % PC 5, Energie 0.13836 % PC 6, Energie 0.052966 %



PC 7, Energie 0.015681 %

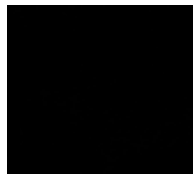


Figure II. 9 L'énergie sur les sept plans de l'image de Lac Mono après l'application de l'algorithme de PCA. PC P (P=1 :7): la projection des données en composantes principales de plan P de l'image satellite.

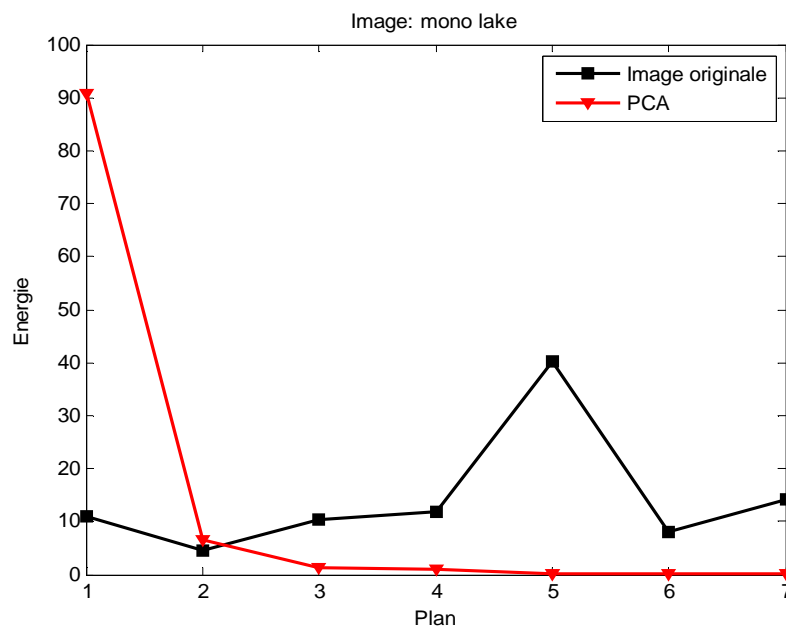


Figure II. 10 L'énergie sur les sept plans de l'image de la base de données Lac Mono avant et après l'application de l'algorithme de PCA.

II.5. Conclusion

Dans ce chapitre, nous avons présenté une analyse détaillée en terme de distribution énergétique de chaque plan a été réalisée dans le but de trouver la meilleure projection des données dans un autre espace pour la compression des images multispéctrale. Donc, il en résulte que nous pouvons concentrer la majorité de l'information dans les deux premiers plans avec l'utilisation de l'algorithme de l'analyse en composante principale (PCA).

Par conséquent, dans le chapitre III, on confirmera le choix d'utilisé la projection en composantes principales qui basé sur la distribution énergétique de l'espace originale le mieux adapté à la compression des images multispéctrale.

Chapitre III

Compression des images satellites

III.1. Introduction

Le développement des applications informatiques s'est accompagné d'un accroissement énorme de l'utilisation d'images, notamment dans le domaine des images satellites. Les images posent, par leur taille importante, de nombreux problèmes quant à leur transmission et à leur stockage.

A la fin des années 80, deux importants groupes de normalisation, le CCITT (Consultative Committee for International Telegraph and Téléphone) et l'ISO (International Standards Organisation) décidèrent de créer, une norme internationale pour la compression d'images fixes. La mise en place d'un standard international était devenue nécessaire pour archiver ou pour faciliter l'échange des images dans des domaines aussi variés que les photos satellites, l'imagerie médicale, la télécopie couleur ou la cartographie... C'est ainsi que fut créé le groupe JPEG (Joint Photographic Experts Group) à l'origine de la norme qui porte son nom. Le codage JPEG est aujourd'hui largement utilisé dans les secteurs de l'informatique et de la communication (appareils photo numériques, scanners, imprimantes, télécopieurs...).

III.2. Principaux algorithmes de compression

Selon le type de la transformée utilisée et suivant le codage, on distingue quelques algorithmes. Parmi lesquels on va citer le standard JPEG, et ensuite, on présente l'approche proposée pour la compression des images satellites.

III.2.1. Standard JPEG

JPEG est le sigle de Joint Photographic Expert Group. Qui édifier une norme du même nom pour la compression des images fixes. Ce standard utilise un protocole en trois étapes principales:

- ❖ Transformation en cosinus discrète (DCT).
- ❖ Quantification.
- ❖ Codage de Huffman.

Le principe de l'algorithme JPEG pour une image en niveaux de gris (une image couleur est décomposée en un plan et deux plans chromatiques qui sont traités de manière identique) est le suivant : La matrice des pixels de l'image est décomposée séquentiellement de gauche à droite et de haut en bas en blocs de 8×8 pixels. Tous les blocs subissent le même traitement. Une transformée en cosinus discret bidimensionnel (DCT pour Discret Cosine Transform) est réalisée sur chaque bloc. Cette transformation concentre l'information de l'image en haut et à

gauche de la matrice, correspondant aux basses fréquences. Les coefficients de la transformée sont ensuite quantifiés à l'aide d'une table de 64 éléments correspondant à un facteur de qualité [11, 13, 22].

La perte de qualité se situe principalement dans l'étape de quantification et le sous échantillonnage des chrominances.

La phase finale est codages sans perte (algorithme de Huffman) sont ensuite réalisés en utilisant les propriétés statistiques des images. Pour cela, on commence par ordonner les coefficients suivant un balayage en zigzag pour placer d'abord les coefficients correspondant aux fréquences les plus basses, qui donne une suite de symboles. Les codes les plus fréquents seront remplacés par d'autres codes comportant un nombre de bits le plus petit possible.

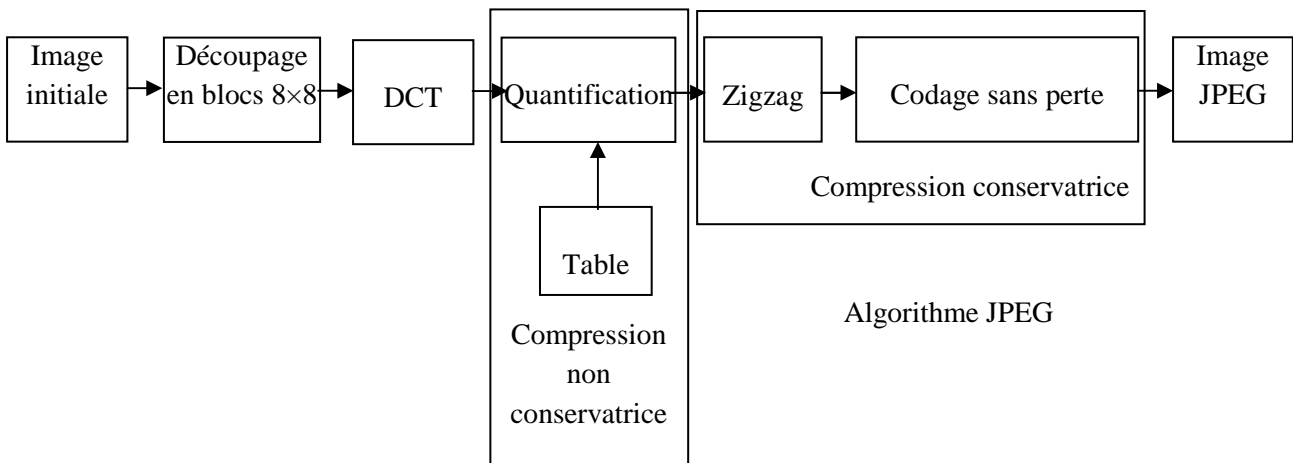


Figure III. 1 Principe de la compression de l' norme JPEG.

III.2.1.1. Transformée en cosinus discrète (DCT et IDCT)

La DCT est l'une des transformées les plus utilisées pour la compression d'image, adoptée dans de nombreux standards de compression. Elle est très similaire à la transformée de Fourier, excepté le fait qu'elle n'utilise pas de valeurs complexes.

Chaque composante de l'image $f(i, j)$ est découpée en blocs de 8x8 pixels, auxquels on applique la DCT pour obtenir les coefficients fréquentiels $F(u, v)$. L'étape de DCT permet de représenter l'image dans une nouvelle base plus propice à la compression. Dans cette représentation, l'information est plus concentrée.

III.2.1.2. Calcul de la DCT et de l'IDCT

Chaque bloc 8×8 est soumis à une transformation par DCT (Discrete Cosine Transform). Le premier coefficient de la DCT est le DC (Direct Component), est proportionnel à la moyenne des valeurs du bloc. Les 63 autres coefficients sont appelés AC (Alternative Component). Ce nouveau domaine transformé permet une décorrélation très forte de l'information. Sur ce bloc de coefficients, les énergies sont groupées en basse, moyenne et haute fréquences.

Par l'application de la DCT, chaque colonne est une fonction cosinus de fréquence différente. La variance est alors concentrée sur les composantes de basse fréquence, et les composantes de haute fréquence seront annulées par quantification.

La DCT travaille sur une matrice carrée représentant un bloc de l'image globale $Img(x,y)$. Le nombre d'échantillons sur l'axe des x doit donc être égal au nombre d'échantillons sur l'axe des Y .

Chaque bloc est de taille $(N \times N)$ 8 par 8 pixels préconisés dans la norme) [22]. La DCT et la IDCT (la transformation en cosinus discrète inverse) fournissent, intuitivement, chacune une matrice carrée. Leurs formules sont données ci-dessous :

$$DCT(i, j) = \frac{1}{\sqrt{2N}} c(i)c(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Img(x, y) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right), \quad (\text{III.1})$$

$$IDCT(i, j) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} c(i)c(j) DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right), \quad (\text{III.2})$$

$$\text{avec } c(i) = \begin{cases} \frac{1}{\sqrt{2}} & \text{si } i=0 \\ 1 & \text{sinon} \end{cases}$$

Il est clair que l'application de la IDCT sur tous les blocs DCT mène à avoir l'image Img . Les formules (III.1) et (III.2) montrent que pour calculer la DCT en un point, il est indispensable de parcourir toutes les valeurs du bloc relatif. La double somme oblige, d'un point de vue algorithmique, en effectuer N^2 calculs de termes pour la DCT et IDCT.

III.2.1.3. Quantification

La quantification est la phase non conservatrice du processus de compression JPEG (excepté les arrondis effectués). Elle amène, moyennement à une diminution de la précision de l'image, mais aussi à réduire le nombre de bits nécessaires au stockage. Pour cela, elle réduit

chaque valeur de la matrice DCT en la divisant par un nombre (quantum) fixé par une table (matrice 8×8) (figure III.2, III.3) de quantification suivant l'équation (III.3) :

$$\text{Valeur quantifiée}(i,j) = \frac{\text{Valeur DCT}(i,j)}{\text{quantum}(i,j)}, \quad (\text{III.3})$$

Cette valeur est arrondie à l'entier le plus proche. Ultérieurement, lors de la reconstruction de l'image, il suffira d'effectuer l'opération inverse ou la déquantification, en multipliant chaque valeur de la matrice quantifiée par le quantum correspondant, pour retrouver une matrice DCT déquantifiée, à partir de laquelle sera établie la matrice des pixels de l'image reconstruite.

Les nombreux tests réalisés ont conduit à retenir en pratique des facteurs de qualité compris entre 1 (l'image reste excellente) et 25 (dégradation encore acceptable).

L'organisme de standardisation ISO a développé un ensemble standard de valeurs de quantifications utilisables par les programmeurs de code JPEG. Ce choix a été rendu possible grâce aux tests intensifs des matrices via des observateurs.

Voici celles préconisées par la norme JPEG (Figure III.2 pour la luminance et Figure III.3 pour les chrominances) :

| | | | | | | | |
|----|----|----|----|-----|-----|-----|-----|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

Figure III. 2 Matrice de quantification pour la luminance (recommandation JPEG).

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 17 | 18 | 24 | 47 | 99 | 99 | 99 | 99 |
| 18 | 21 | 26 | 66 | 99 | 99 | 99 | 99 |
| 24 | 26 | 56 | 99 | 99 | 99 | 99 | 99 |
| 47 | 66 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |
| 99 | 99 | 99 | 99 | 99 | 99 | 99 | 99 |

Figure III. 3 Matrice de quantification pour les chrominances (recommandation JPEG).

Habituellement, la matrice de quantification Q est obtenue à partir de formules plus ou moins simples, permettant de “choisir” la perte de qualité. La formule suivante est un exemple qui donne de bons résultats :

$$Q = q(i, j) \text{ avec } q(i, j) = 1 + K(1 + m(i + j))$$

Avec i l'indice de ligne, j l'indice de colonne, m une constante (souvent égale à 1) et K le facteur de qualité (choisi entre 1 et 25).

III.2.2. Méthode proposée

On présente dans cette partie l'approche proposée pour la compression des images satellites. Cette compression et reconstruction des images multispectrale est présentée dans les Figures III.4 et Figure III.5, avec une projection d'espace originale vers PCA, cette projection permet une compression plus efficace car l'information est principalement condensée dans le première plan.

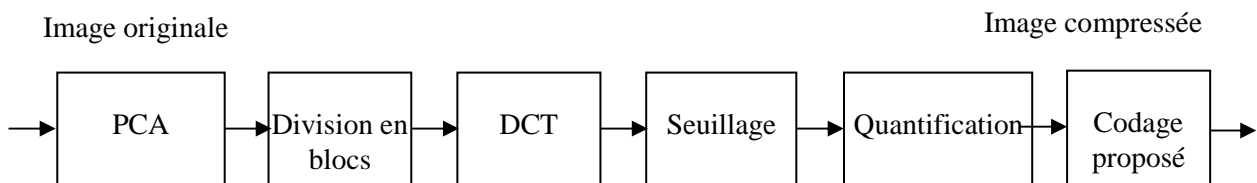


Figure III. 4 Schéma globale de la compression des images multispectrales.

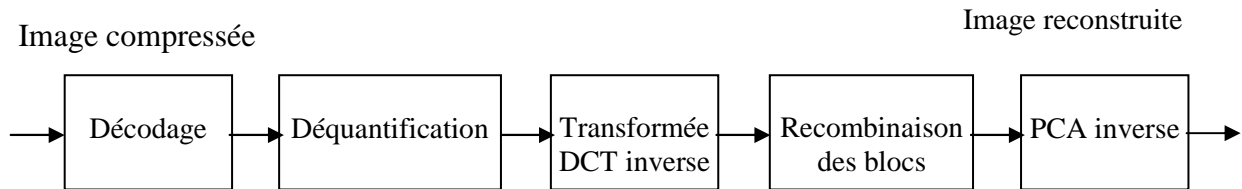


Figure III. 5 Schéma globale de reconstruction des images satellites.

Les différentes étapes de notre méthode proposée :

- Une soustraction de la valeur moyenne sur chaque plan est appliquée sur l'image multispectrale originale, ceci mène à une image reconstruite à valeur moyenne nulle.
- Division en blocs, l'objectif de cette étape est de limiter le nombre de pixels à traiter à la fois pour diminuer le temps de calcul et la complexité des circuits électroniques. La dimension des blocs est variable selon les méthodes. Dans notre cas nous avons utilisé deux tailles des blocs 8×8 ou 16×16 .
- La transformation en DCT : on utilise l'équation III.1 pour chaque bloc, ce qui permet de transformer les pixels d'un bloc 8×8 (16×16) d'une image en un autre bloc de 8×8 (16×16) contenant les composantes fréquentielles correspondantes. On trouve des coefficients qui caractérisent les basses fréquences et d'autres qui caractérisent les hautes fréquences. L'utilisation de la DCT permet de concentrer l'énergie dans les basses fréquences.
- Seuillage : si les coefficients obtenus après la DCT est inférieurs ou égaux à TH, ces coefficients mis à zéro, sinon reste même valeurs de XDCT. Plus le seuil TH est élevé, plus le taux de compression est grand, mais une distorsion d'image importante est inévitable (on varie le seuil TH de 5 jusqu'à 50, TH=5, 10, 25, 35, 50).
- Quantification : les vecteurs de coefficients non nuls sont en suite quantifiés par un quantificateur linéaire de différents nombre de bits (7, 8, 9 bits), l'objectif de cette quantification est de réduire le nombre de bits nécessaires à la représentation des coefficients pour codage de l'image multispectrale. La quantification QDCT des coefficients XDCT est effectuée selon l'équation suivante :

$$QDCT = \left\lfloor 1 + \frac{XDCT - XDCT_{\min}}{XDCT_{\max} - XDCT_{\min}} \times (2^p - 2) \right\rfloor \quad (III.4)$$

Pour un nombre de bits égal a 7 bits QDCT est contient des valeurs sur un intervalle entre 1 et 127.

$\lfloor \rfloor$ représente l'arrondi des valeur de QDCT.

$XDCT_{min}$ valeur minimale de $XDCT$.

$XDCT_{max}$ valeur maximale de $XDCT$.

p nombre de bits de quantification.

La déquantification se fait selon :

$$X_{dq} = \frac{QDCT - 1}{(2^p - 2)} \times (XDCT_{max} - XDCT_{min}) + XDCT_{min} \quad (III.5)$$

Où X_{dq} est le coefficient reconstruit.

- Codage sans perte : pour chaque bloc de 8×8 ou 16×16 pixels on ordonne les coefficients de QDCT suivant le balayage zigzag, ensuite, on applique le codeur TRE (Two-Role Encoder) [30]. Nous avons proposée un bloc remplace le bloc 8×8 ou 16×16 par un train de bits (Figure III.6), ce bloc est composé de (A, PNZ, L), où (A) est l'activation du bloc de 1 bits, (PNZ) position de la dernière valeur non nulle, avec $PNZ = NZ - 1$, où NZ est le nombre de la dernière valeur non nulle. Le nombre de bits pour représenter (PNZ) est de 6 bits dans le cas division en 8×8 pixels, de 8 bits pour le deuxième cas (16×16 pixels). L est codage des non zéros selon le nombre de bits choisi (7, 8, 9 bits). L'opération et le codage par l'algorithme de Huffman.

| | | |
|-------------|--------------------|--------------------------------------|
| A | PNZ | L |
| 1bit | 6 ou 8 bits | $p \times NZ$ bits |

Figure III. 6 Bloc du codage sans perte.

Pour la reconstruction des images multispectrale on applique l'opération inverse : le décodage, la déquantification, et la transformée en cosinus discret inverse.

Pour la mesure de qualité de notre méthode proposée, plusieurs critères de mesures ont été proposée dans la littérature, dans notre cas en utilise trois critères d'évaluation : *PSNR*, *CR*, *bpp* après l'addition de la valeur moyenne précédemment retranchée aux images originales et reconstruites.

PSNR (Peak Signal to Noise Ratio) est une mesure de distorsion défini par :

$$PSNR = 10 \times \log_{10} \frac{d^2}{MSE} \quad (III.6)$$

Où d est la valeur maximum possible pour un pixel.

MSE est l'erreur quadratique moyenne est définie pour deux images, image originale (I_{orig}) et l'image reconstruite (I_{rec}).

$$MSE = \frac{1}{P \times N \times M} \sum_{p=1}^P \sum_{i=1}^N \sum_{j=1}^M (I_{orig}(i, j, p) - I_{rec}(i, j, p))^2 \quad (III.7)$$

P : nombre de plan de l'image satellite, et $N \times M$ est la taille de l'image.

Le taux de compression est donné par la relation suivante:

$$CR = \frac{\text{nombre de bits de l'image multispectrale originale}}{\text{image compressée en bits}} \quad (III.9)$$

Le nombre de bit par pixel est défini comme suit:

$$bpp = \frac{Nb \times P}{CR} \quad (III.10)$$

Nb est nombre de bits, et P nombre de plans d'image multispectrale.

III.2.3 Résultats sur les images satellites

La simulation numérique est une étape importante dans l'étude et la mise en oeuvre de traitement et compression d'image. En effet, elle permet, d'une part, la validation des études théoriques et, d'autre part, l'optimisation du système par l'analyse de l'impact des divers paramètres sur les performances.

Afin de tester l'efficacité de notre méthode, nous avons utilisé deux images multispectrales: "Khenchela", et "Lac Mono" de taille 816×1154 et 512×512 , respectivement.

Dans les figures suivantes (Figure III.7 a III.11), nous présentons les différents résultats de notre stratégie proposée dans ce travail.

Deux différentes tailles de bloc (8×8 , 16×16) ont été testées sur des images compressées pour plusieurs taux de compression (on fait varier le seuil entre 5 et 50) et ainsi pour trois nombres de bits (7, 8, 9 bits).

Ensuite on calcule le $PSNR$, CR et bpp à la fois dans l'espace originale et dans l'espace des données projetées PCA (Figure III.7 et Figure III.8). Ces deux dernières figures montrent la supériorité des performances de la technique proposée lorsque travaillant dans l'espace PCA que dans l'espace originale.

Puis on regarde l'influence de la méthode proposée et codeur Huffman pour différentes valeurs de seuil, taille de blocs, nombre de bits. Les résultats obtenus pour l'image de kenchela et Lac Mono sont présentés sur les Figure III.9 et Figure III.10. La méthode PCA+DCT+sans perte+Huffman son taux de compression est meilleur que celui de la version précédente (PCA+DCT+sans perte).

En terminant ensuite par une étude comparative entre la meilleure stratégie de notre méthode et la norme JPEG [31] (voir la Figure III.11).

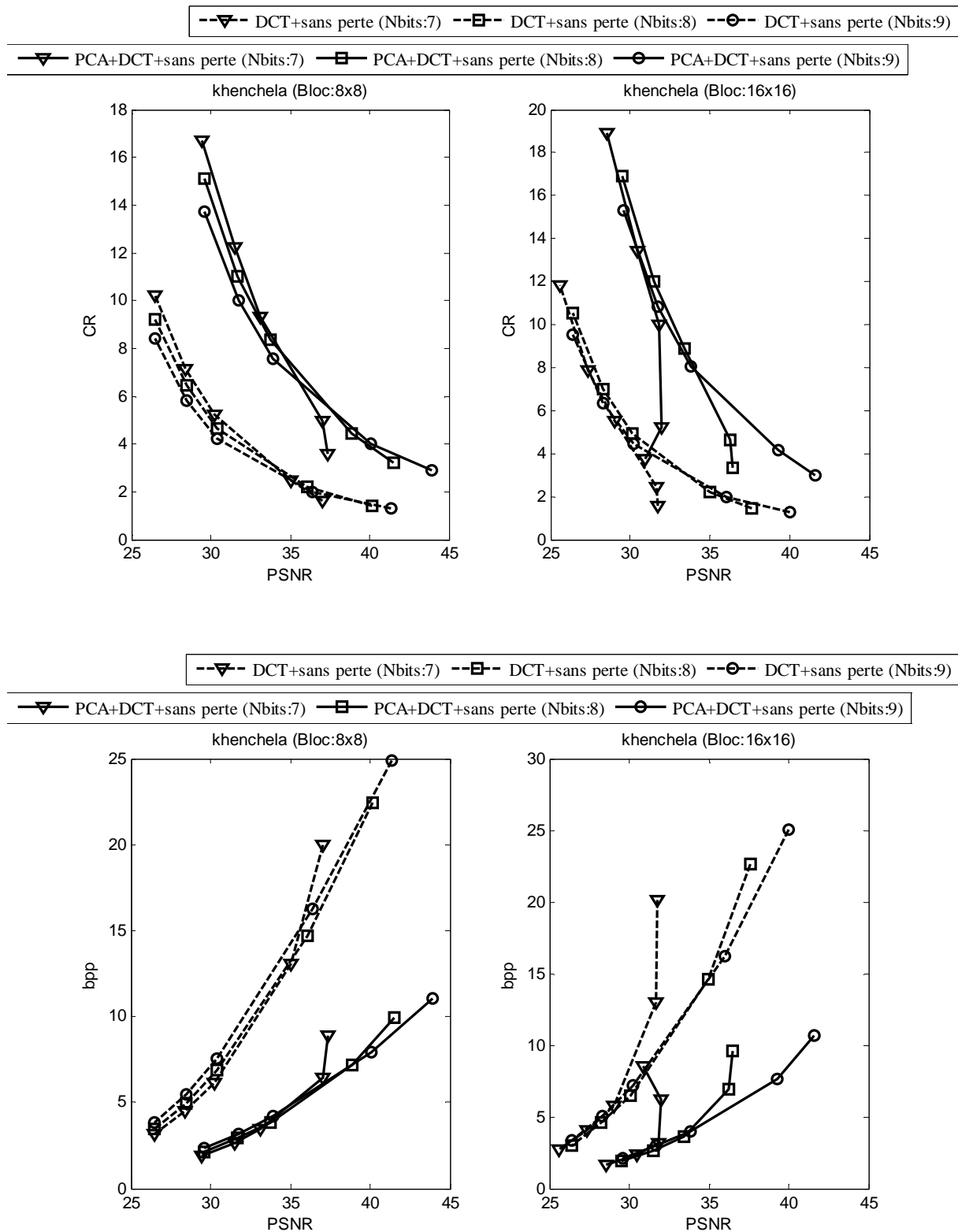


Figure III. 7 Evaluation du taux de compression (CR) et nombre de bit par pixel (bpp) en fonction de PSNR entre la méthode proposée (DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l’application de PCA sur l’image de la ville de Khenchela, pour les nombres de bits 7,8 et 9bits.

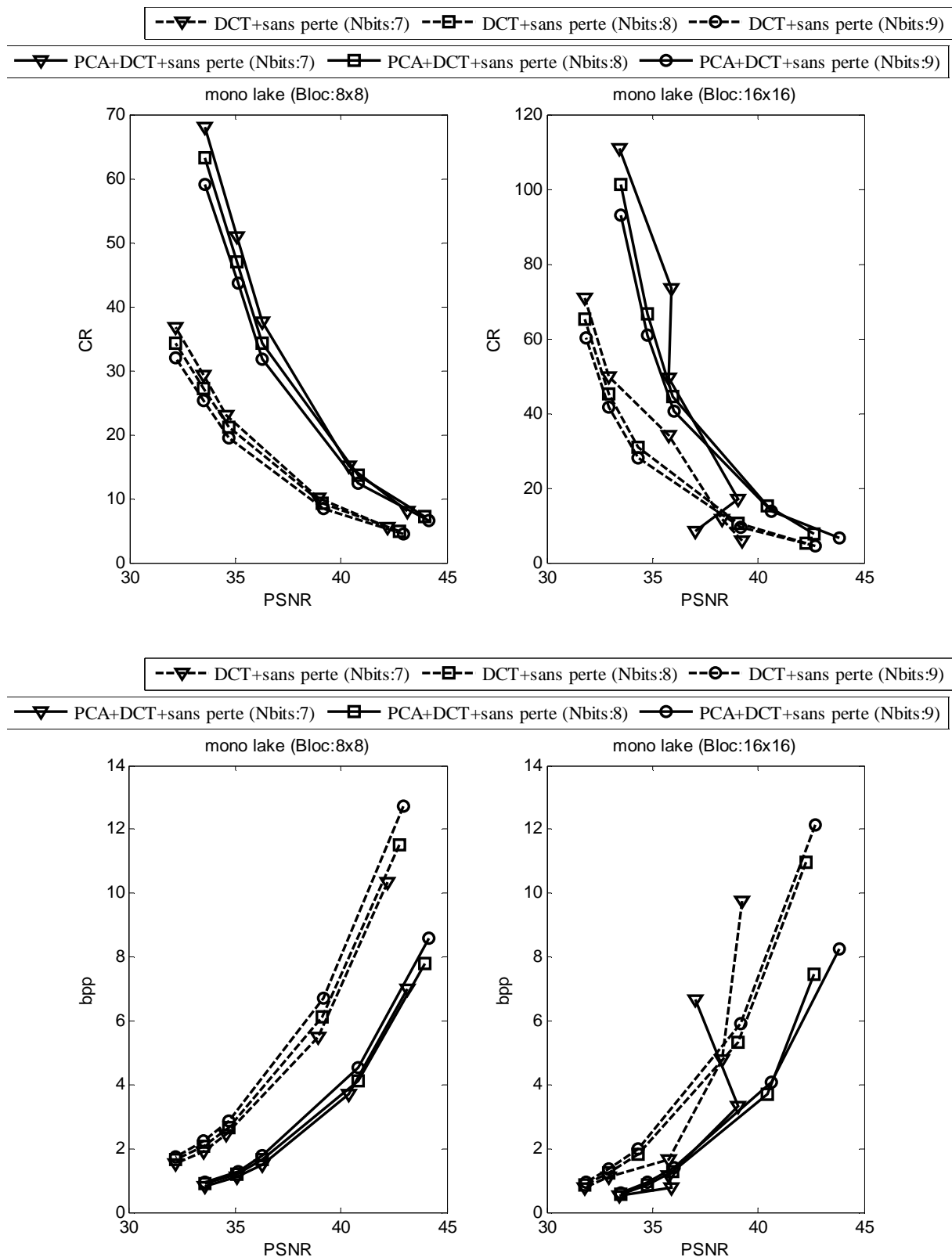


Figure III. 8 Evaluation du taux de compression (CR) et nombre de bit par pixel (bpp) en fonction de $PSNR$ entre la méthode proposée (DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de PCA sur l'image de Lac Mono, pour les nombres de bits: 7, 8, 9 bits.

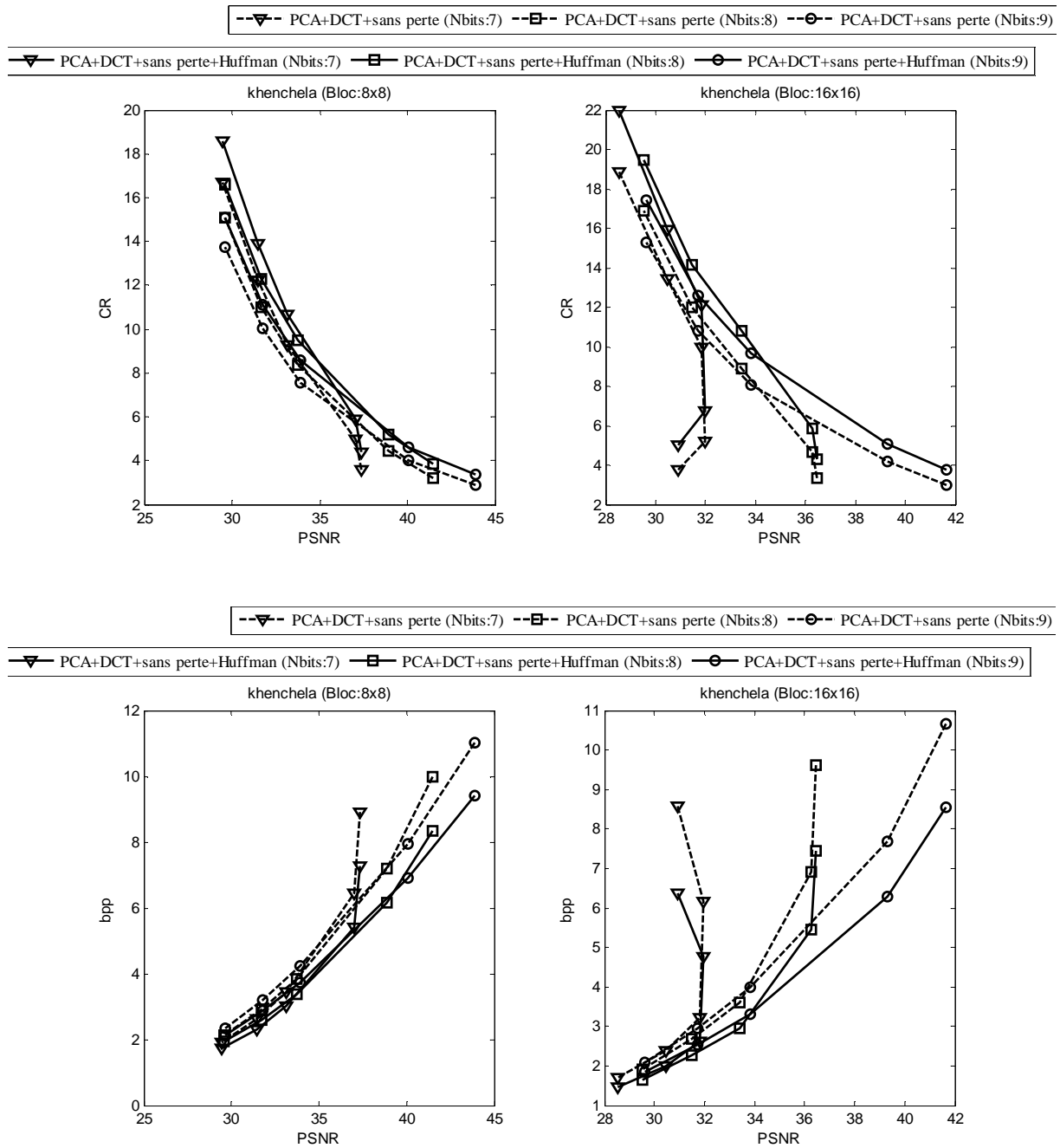


Figure III. 9 Evaluation du taux de compression (CR) et nombre de bit par pixel (bpp) en fonction de $PSNR$ entre la méthode proposée (PCA+DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de codeur Huffman sur l'image de la ville de Khenchela, pour les nombres de bits: 7, 8, 9 bits.

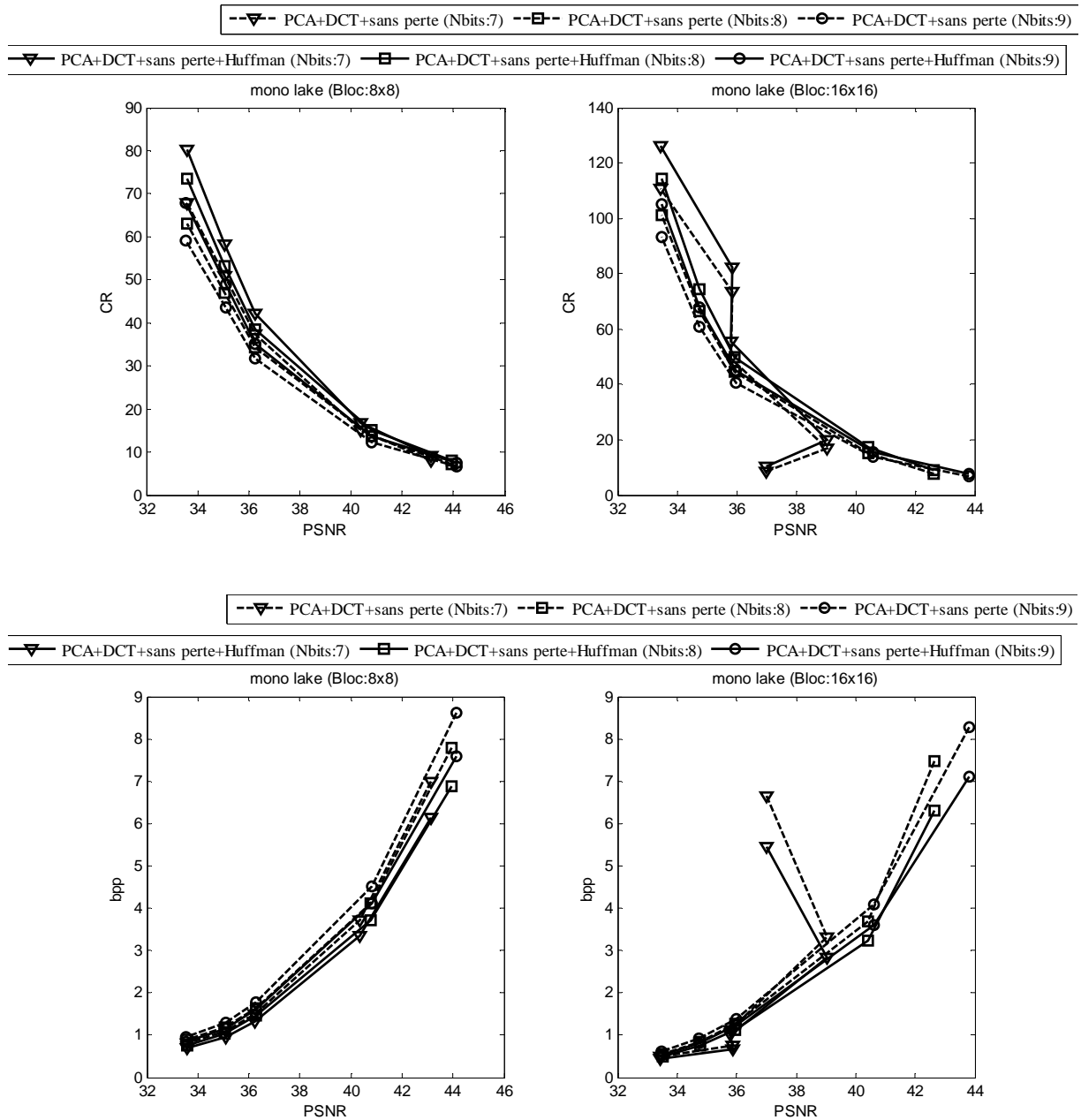


Figure III. 10 Evaluation du taux de compression (CR) et nombre de bit par pixel (bpp) en fonction de $PSNR$ entre la méthode proposée (PCA+DCT+codage sans perte, Bloc 8×8 et 16×16) avant et après l'application de codeur Huffman sur l'image de Lac Mono, pour les nombres de bits: 7, 8, 9 bits.

Pour valider notre travail, il est intéressant de comparer les meilleurs résultats de notre méthode (PCA+DCT+sans perte+Huffman, Bloc 16×16, nombre de bits : 9, TH=5 ,10 ,25 ,35 ,50 ,60 ,70 ,80 ,90 ,100) avec ceux la méthode JPEG et PCA+JPEG. Cette comparaison est illustrée dans la Figure III.11.

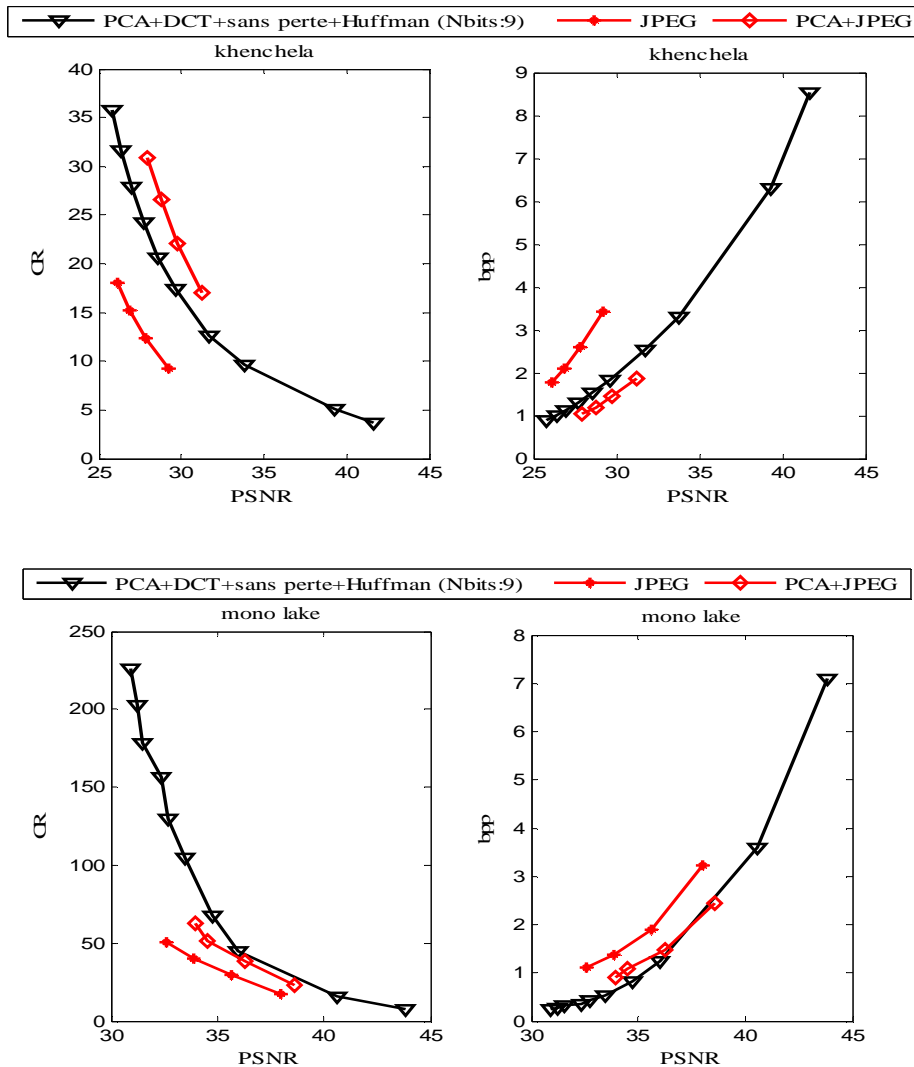





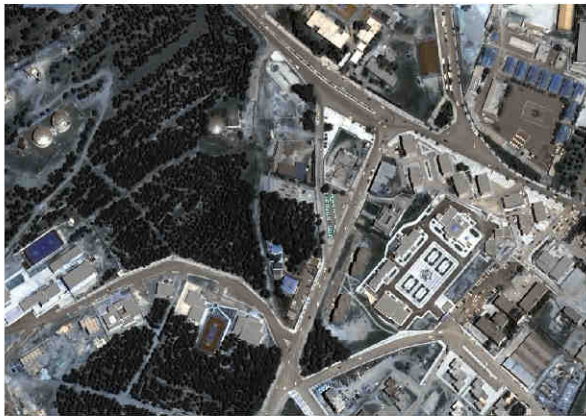




Figure III. 11 Evaluation du taux de compression (CR) et nombre de bit par pixel (bpp) en fonction de $PSNR$ entre la méthode proposée (PCA+DCT+codage sans perte+Huffman, Bloc 16×16, nombre de bits: 9 bits), la norme JPEG, et PCA+JPEG, sur les deux images multispectrale : Khenchela et Lac Mono.

Dans les Figures III.12 à III.15, nous comparons par une évaluation subjective la compression des images de la ville khenchela et Lac Mono obtenues par notre méthode avec la norme PCA+JPEG, ceci est effectué pour deux taux de compression différents (le seuil est 5 et 50), avec deux types de division de blocs (8×8, 16×16), et on varie le nombre de bits : 7, 8 et 9 bits.

| Khenchela | TH N_bits | 5 | 50 |
|-----------|--------------|---|---|
| Blocs 8×8 | 7 | <p>Blocs 8×8, Nbits= 7, TH= 5, PSNR= 37.3494, CR= 3.592, CR-Huff= 4.4105</p>  | <p>Blocs 8×8, Nbits= 7, TH= 50, PSNR= 29.4645, CR= 16.7035, CR-Huff= 18.5607</p>  |
| | 8 | <p>Blocs 8×8, Nbits= 8, TH= 5, PSNR= 41.5056, CR= 3.2097, CR-Huff= 3.8323</p>  | <p>Blocs 8×8, Nbits= 8, TH= 50, PSNR= 29.5892, CR= 15.0799, CR-Huff= 16.5736</p>  |

| | | | |
|-------------|---|--|---|
| | 9 | <p>Blocs 8x8, Nbits= 9, TH= 5, PSNR= 43.9056, CR= 2.901, CR-Huff= 3.3946</p>  | <p>Blocs 8x8, Nbits= 9, TH= 50, PSNR= 29.6158, CR= 13.7439, CR-Huff= 15.0612</p>  |
| Blocs 16x16 | 7 | <p>Blocs 16x16, Nbits= 7, TH= 5, PSNR= 30.9048, CR= 3.738, CR-Huff= 5.0256</p>  | <p>Blocs 16x16, Nbits= 7, TH= 50, PSNR= 28.5205, CR= 18.8822, CR-Huff= 21.9766</p>  |

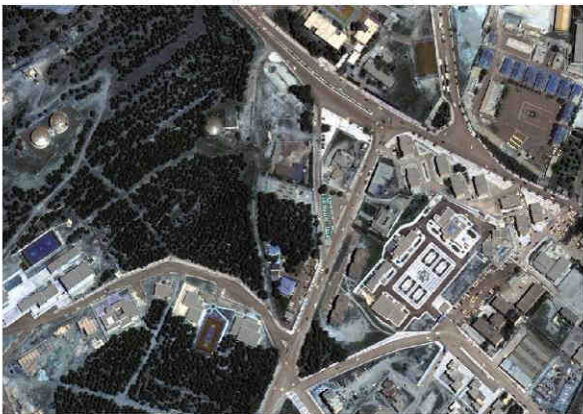
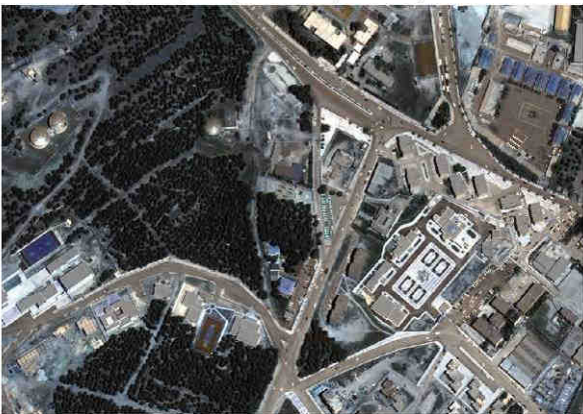
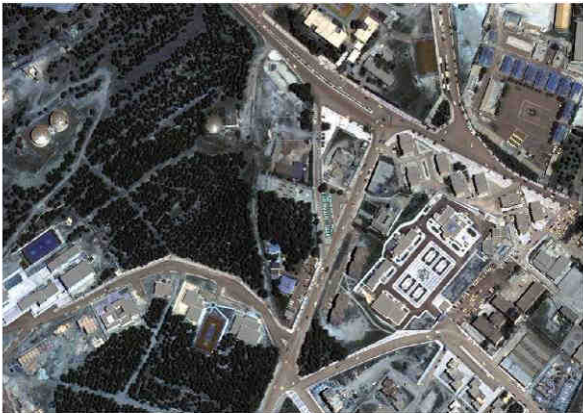

| | | |
|---|---|---|
| 8 | <p>Blocs 16×16, Nbits= 8, TH= 5, PSNR= 36.4617, CR= 3.3287, CR-Huff= 4.3036</p>  | <p>Blocs 16×16, Nbits= 8, TH= 50, PSNR= 29.5281, CR= 16.8944, CR-Huff= 19.4484</p>  |
| 9 | <p>Blocs 16×16, Nbits= 9, TH= 5, PSNR= 41.6304, CR= 3.0002, CR-Huff= 3.7395</p>  | <p>Blocs 16×16, Nbits= 9, TH= 50, PSNR= 29.6065, CR= 15.2853, CR-Huff= 17.4259</p>  |

Figure III. 12 Compression d'une image de Khenchela (RVB) par l'approche proposée.

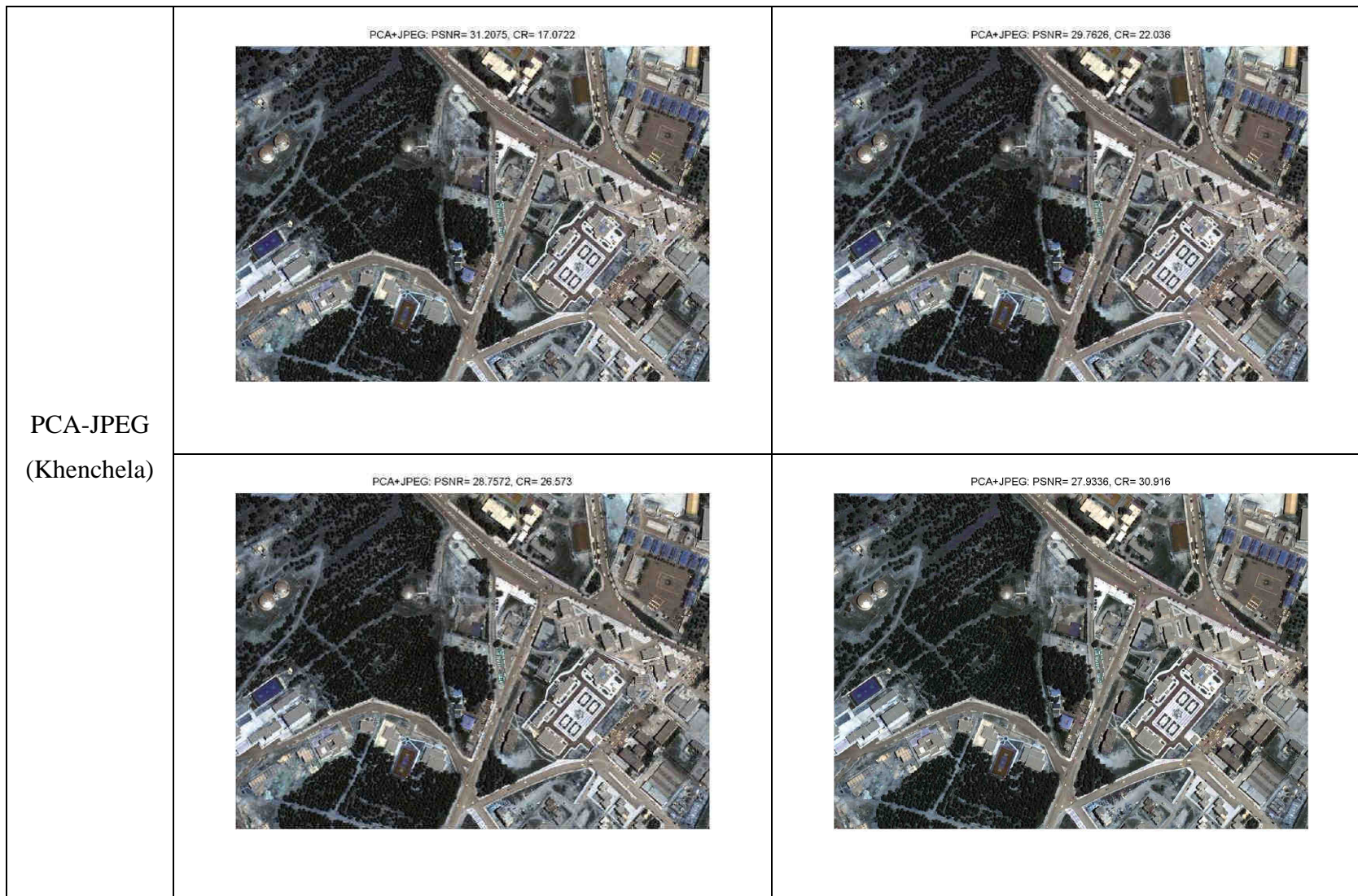
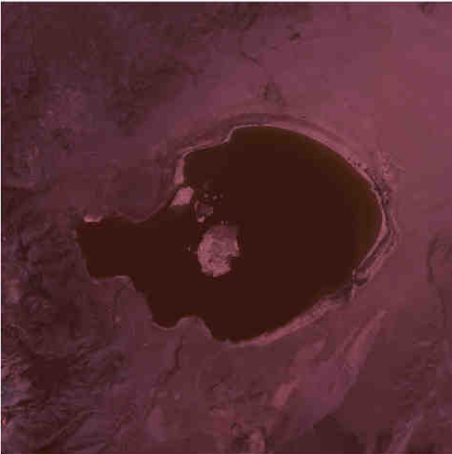

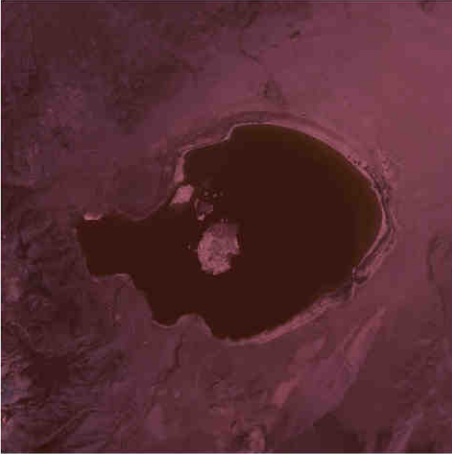

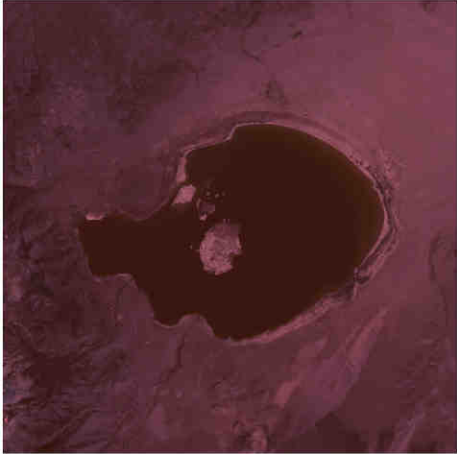

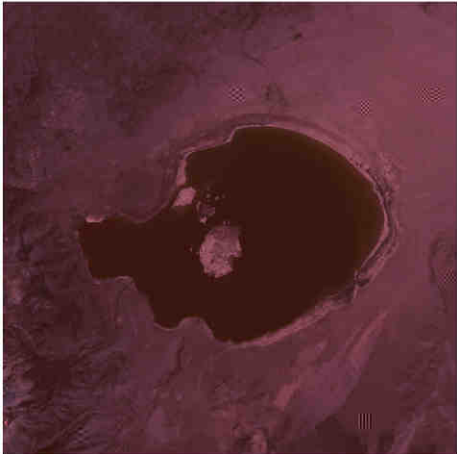
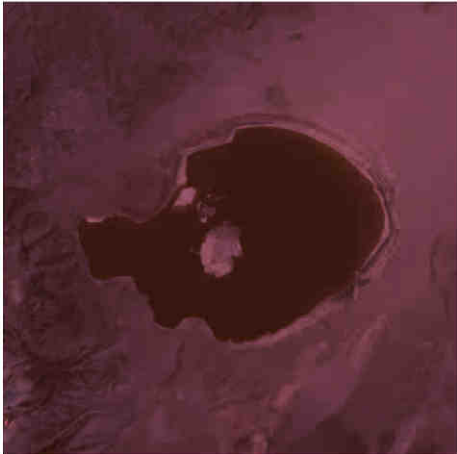


Figure III. 13 Compression d'une image de Khenchela (RVB) par l'algorithme PCA-JPEG.

| Lac Mono | TH N_bits | 5 | 50 |
|-----------|--------------|---|---|
| Blocs 8×8 | 7 | <p>Blocs 8×8, Nbits= 7, TH= 5, PSNR= 43.1165, CR= 7.9975, CR-Huff= 9.1384</p>  | <p>Blocs 8×8, Nbits= 7, TH= 50, PSNR= 33.5695, CR= 67.979, CR-Huff= 80.1615</p>  |
| | 8 | <p>Blocs 8×8, Nbits= 8, TH= 5, PSNR= 43.9364, CR= 7.1762, CR-Huff= 8.1467</p>  | <p>Blocs 8×8, Nbits= 8, TH= 50, PSNR= 33.5888, CR= 63.1724, CR-Huff= 73.5198</p>  |

| | | | |
|-------------|---|--|---|
| | 9 | <p>Blocs 8x8, Nbits= 9, TH= 5, PSNR= 44.1344, CR= 6.5078, CR-Huff= 7.3862</p>  | <p>Blocs 8x8, Nbits= 9, TH= 50, PSNR= 33.5482, CR= 59.0006, CR-Huff= 67.7816</p>  |
| Blocs 16x16 | 7 | <p>Blocs 16x16, Nbits= 7, TH= 5, PSNR= 37.0093, CR= 8.4105, CR-Huff= 10.2566</p>  | <p>Blocs 16x16, Nbits= 7, TH= 50, PSNR= 33.4308, CR= 110.7829, CR-Huff= 126.3867</p>  |

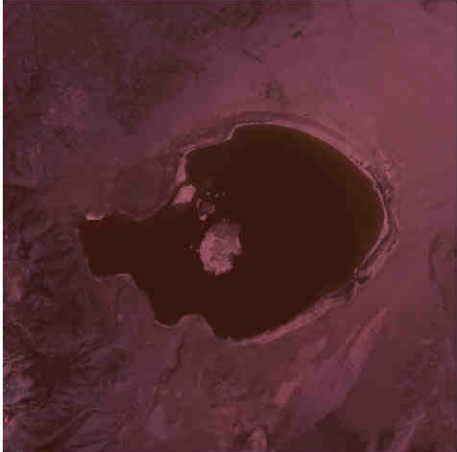
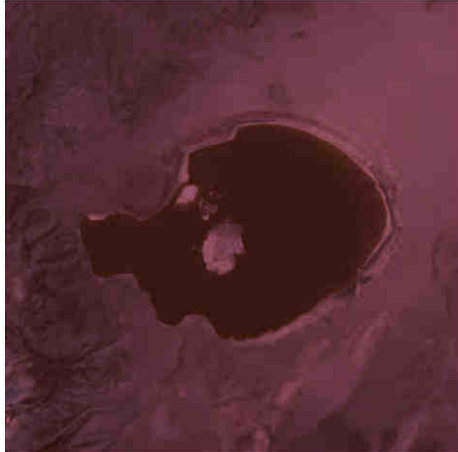
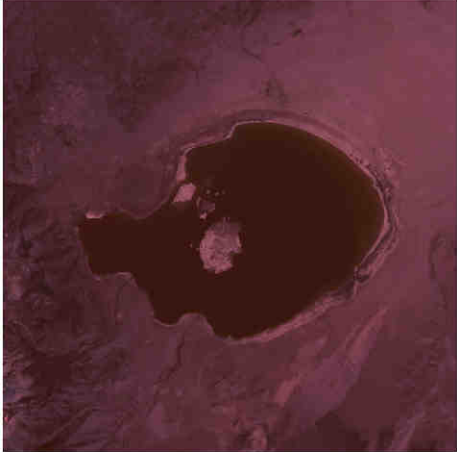
| | | |
|---|--|--|
| 8 | <p>Blocs 16×16, Nbits= 8, TH= 5, PSNR= 42.6443, CR= 7.5031, CR-Huff= 8.8844</p>  | <p>Blocs 16×16, Nbits= 8, TH= 50, PSNR= 33.51, CR= 101.1497, CR-Huff= 114.4279</p>  |
| | 9 | <p>Blocs 16×16, Nbits= 9, TH= 5, PSNR= 43.8242, CR= 6.7724, CR-Huff= 7.8769</p>  |

Figure III. 14 Compression d'une image de Lac Mono (RVB) par l'approche proposée.

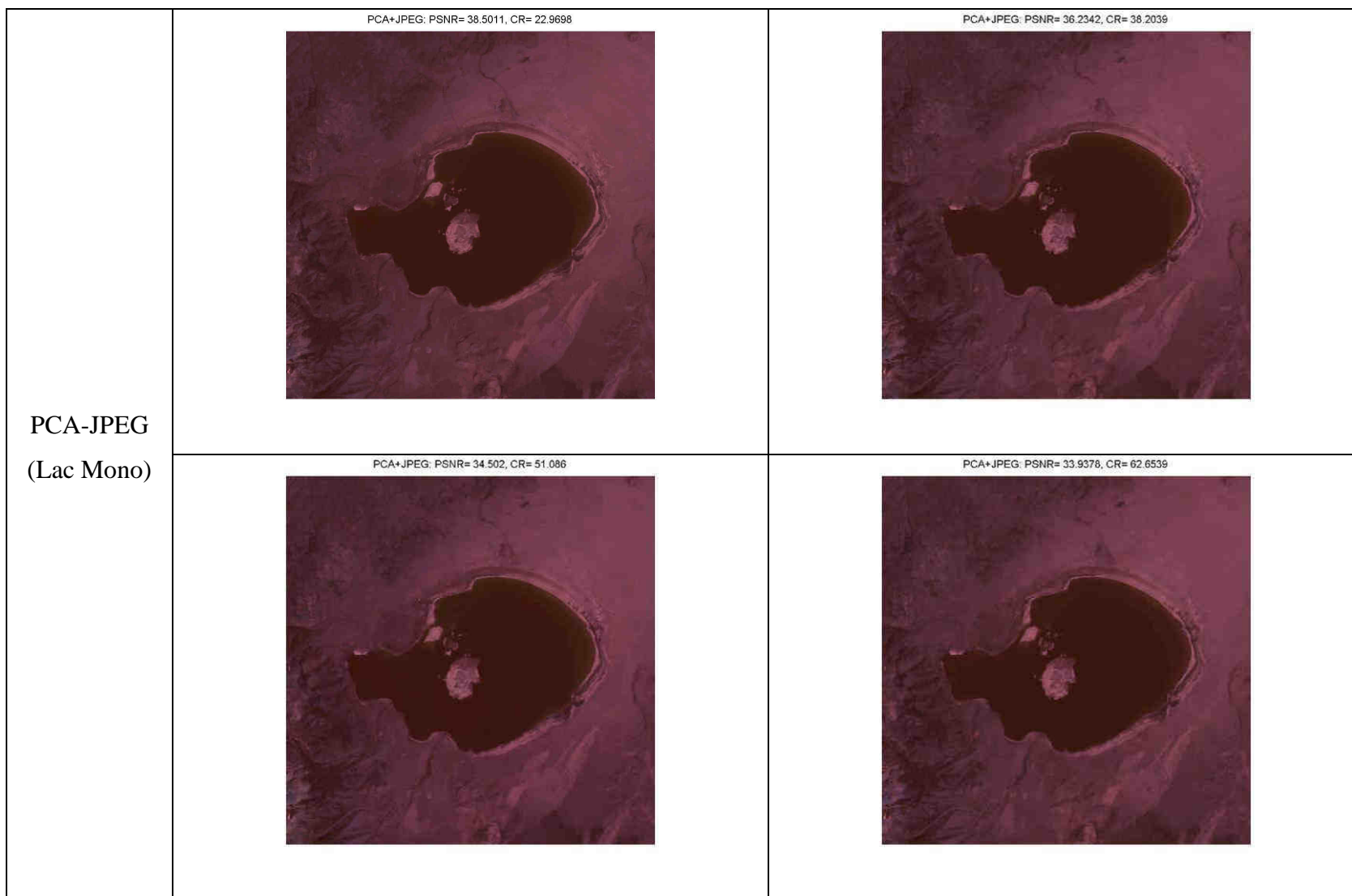


Figure III. 15 Compression d'une image de Lac Mono (RVB) par l'algorithme PCA-JPEG.

Les résultats obtenus sur deux images multispectrales montrer de bonnes capacités des stratégies proposées pour la compression des images satellites. En général, la méthode proposée basé sur la combinaison des trois parties (PCA+DCT+sans perte+Huffman): le prétraitement (changement d'espace originale vers un espace en composante principale PCA), la transformation DCT, la compression sans perte, le codeur Huffman, sont caractérisés par des performances supérieures en termes de qualité (PSNR), de taux de compression (CR), et nombre de bits par pixel (*bpp*), par rapport à la norme JPEG.

L'application de l'analyse en composante principale sur la norme JPEG donne des meilleurs résultats dans le cas de la base de données Khenchela par apport à notre méthode, nous pouvons voir que le taux de compression obtenu par la méthode proposée (PCA+DCT+sans perte+Huffman) pour les deux images multispectrale est supérieur à celui de JPEG et PCA-JPEG et ceci est évidemment en termes *PSNR* en moyenne (Figure III.11).

Pour une lisibilité plus facile de ces résultats, nous calculons la moyenne de PSNR et CR sur les différents valeurs de seuil (voir tableau III.1), le critère de qualité (PSNR) et taux de compression obtenus sur les deux images de Khenchela et Lac Mono de la méthode (PCA+DCT+sans perte+Huffman) pour un nombre de représentation des pixels de 9 bits et un seuil entre 5 et 50, donnera un PSNR de 35.8563, 37.9760 et *CR* de 8.5477, 34.5687, respectivement, pour la première type de découpage 8×8. Pour la taille de bloc 16×16, on obtient un PSNR de 35.2171, 37.7283, et un *CR* de 9.6955, 48.3020, sur les deux images de la ville Khenchela et Lac Mono, respectivement, on fixe les mêmes paramètres précédents.

En observation générale, la qualité des images restaurées obtenues avec la méthode de PCA+DCT+sans perte+Huffman en utilisant le type de découpage de 16×16 et un nombre de bits de 9bits est meilleure pour la reconstruction des images satellites.

| Image multispectrale Méthodes | Khenchela | | Lac Mono | |
|--------------------------------------|----------------|---------------|----------------|----------------|
| | PSNR | CR | PSNR | CR |
| Bloc 8×8 | | | | |
| DCT+sans perte (Nbits:7) | 31.4469 | 5.3267 | 36.2751 | 20.9379 |
| DCT+sans perte (Nbits:8) | 32.3201 | 4.7866 | 36.4453 | 19.3054 |
| DCT+sans perte (Nbits:9) | 32.6425 | 4.3460 | 36.4887 | 17.9124 |
| PCA+DCT+sans perte (Nbits:7) | 33.6905 | 9.3603 | 37.6723 | 35.9048 |
| PCA+DCT+sans perte (Nbits:8) | 35.0948 | 8.4201 | 37.9280 | 33.0565 |
| PCA+DCT+sans perte (Nbits:9) | 35.8563 | 7.6516 | 37.9760 | 30.6317 |
| PCA+DCT+sans perte+Huffman (Nbits:7) | 33.6905 | 10.6914 | 37.6723 | 41.3406 |
| PCA+DCT+sans perte+Huffman (Nbits:8) | 35.0948 | 9.4724 | 37.9280 | 37.6852 |
| PCA+DCT+sans perte+Huffman (Nbits:9) | 35.8563 | 8.5477 | 37.9760 | 34.5687 |
| Bloc 16×16 | | | | |
| DCT+sans perte (Nbits:7) | 29.0786 | 5.8512 | 35.6130 | 34.4487 |
| DCT+sans perte (Nbits:8) | 31.4661 | 5.2221 | 36.0696 | 31.3682 |
| DCT+sans perte (Nbits:9) | 32.2031 | 4.7150 | 36.1924 | 28.7969 |
| PCA+DCT+sans perte (Nbits:7) | 30.7392 | 10.2411 | 36.2275 | 51.7559 |
| PCA+DCT+sans perte (Nbits:8) | 33.4292 | 9.1481 | 37.4616 | 46.9615 |
| PCA+DCT+sans perte (Nbits:9) | 35.2171 | 8.2659 | 37.7283 | 42.9829 |
| PCA+DCT+sans perte+Huffman (Nbits:7) | 30.7392 | 12.3553 | 36.2275 | 58.8615 |
| PCA+DCT+sans perte+Huffman (Nbits:8) | 33.4292 | 10.9161 | 37.4616 | 52.9633 |
| PCA+DCT+sans perte+Huffman (Nbits:9) | 35.2171 | 9.6955 | 37.7283 | 48.3020 |

Tableau III. 1 Les performances mesurées (moyenne PSNR, moyenne CR) en appliquant l'approche sur les images (Khenchela, Lac Mono) reconstituées, les paramètres : deux type de découpage 8×8 et 16×16, varie le nombre de bits (7, 8 et 9bits), TH entre 5 et 50.

III.3. Conclusion

Le but principal de compression des images est de réduire la quantité de bits nécessaire pour décrire tout en gardant un aspect visuel acceptable, des images reconstruites. Pour résoudre ce problème nous avons proposée une méthode de compression basée sur PCA, DCT, la compression sans perte, et codeur Huffman sur les images satellites multispectrale. Nous avons validé la méthode proposée en utilisant deux images satellites de khenchela et Lac Mono.

Pour montrer l'efficacité de notre méthode, nous avons varié plusieurs paramètres : la taille de bloc 8×8 et 16×16, le nombre de bits pour la représentation des pixels (7, 8 et 9bits), le seuillage qui exposé à l'image satellite. Le seuillage permet d'ajouter des distorsions dans les images satellites, plus le seuil TH est élevé, plus le taux de compression est grand, et enfin une méthode hybride entre la méthode proposée sans perte (un vecteur remplace le bloc 8×8 ou 16×16 par un train de bits), codage TRE et codeur Huffman. Tous ces paramètres sont effectués avant et après la projection d'espace originale en composante principale.

D'après la comparaison de la méthode proposée entre eux, nous pouvons conclure que l'utilisation de l'analyse en composante principale, la DCT, compression sans perte et codeur Huffman sur les deux images satellites (Khenchela et Lac Mono) donne de très bons résultats. Les résultats que nous avons obtenus et que nous avons présentés dans ce chapitre sont assez satisfaisants du point de vue de l'amélioration des performances du codeur ($PSNR$, taux de compression Cr , nombre de bits par pixel bpp) par rapport à celui pour la norme JPEG et PCA-JPEG.

Conclusion générale

Conclusion générale

Le travail réalisé dans ce mémoire, c'est l'application de la méthode proposée pour la compression des images satellites en utilisant l'analyse en composante principale, la transformation cosinus discrète (DCT), codage sans perte, codeur Huffman. L'objectif de ce travail est l'amélioration de performance de la mesure de qualité d'image multi-spectrale avec un stockage des données plus compact.

Habituellement, les images couleurs sont stockées sous le format brut. Or l'espace originale n'est pas le mieux adapté pour le processus de compression, puisqu'il y a une forte corrélation entre les différents plans.

Du fait qu'il y a une forte corrélation entre les différents plans de l'espace originale, nous effectuons, dans une première étape, un changement de l'espace originale vers un autre espace de en composante principale moins corrélé. Evidemment, cette transformation permet une compression plus efficace car l'information est principalement concentrée dans le premier plan.

Les résultats que nous avons obtenus, et que nous avons présentés dans ce travail, sont assez satisfaisants du point de vue amélioration de performances du codeur (PSNR, CR, bpp) par rapport à la norme JPEG. En moyenne de taux de compression et PSNR, notre méthode proposée est meilleur par rapport PCA-JPEG.

L'application de l'approche proposée sur deux différentes images satellites (Khenchela et Lac Mono), a montré que : la meilleure performance peut être obtenue avec l'utilisation de PCA, taille de bloc 16×16 , et nombre de bits égal à 9, ces paramètres sont les plus adaptées à notre méthode.

Bibliographie

Bibliographie

- [1] X. Delaunay, Compression d'images satellite par post-transformées dans le domaine ondelettes, thèse doctorat de l'université de Toulouse, 12 novembre 2008.
- [2] S. Poobal, G. Ravindran, Comparison of Compression Ability Using DCT and Fractal Technique on Different Imaging Modalities, World Academy of Science, Engineering and Technology, 12 (2007) 55-60.
- [3] A.C. Bovik, Handbook of image and video processing, Academic press, 2010.
- [4] A. Skodras, C. Christopoulos, T. Ebrahimi, The JPEG 2000 still image compression standard, IEEE Signal Processing Magazine, 18 (2001) 36-58.
- [5] J.M. Shapiro, Embedded image coding using zerotrees of wavelet coefficients, IEEE Transactions on Signal Processing, 41 (1993) 3445-3462.
- [6] B.-J. Kim, W.A. Pearlman, An embedded wavelet video coder using three-dimensional set partitioning in hierarchical trees (SPIHT), in: Data Compression Conference DCC'97. IEEE Proceedings 1997.
- [7] Yu W, F.J. Sun F, Efficient rate control for jpeg-2000, IEEE Transactions on Circuits and Systems for Video Technology, 16 (2006) 577-589.
- [8] Gershikov E, P. M, On color transforms and bit allocation for optimal subband image compression, Signal Processing: Image Communication, 22 (2007) 1-18.
- [9] A. Singh I, gathoklis P, A. A., Lossless compression of color images using an improved integer-based nonlinear wavelet transform, In: Proceedings of 1997 IEEE international symposium on circuits and systems, 1997. ISCAS'97, 1997.
- [10] Dhara B, C. B, Color image compression based on block truncation coding using pattern fitting principle, Pattern Recognition, 40 (2007) 2408-2417.
- [11] A. Boucetta, Etude de l'effet des transformées des décorrélation en compression des images couleurs RGB, Magister en informatique, Université El Hadj Lakhdar de Batna, 2010.
- [12] W.P. Loughlin, Principal component analysis for alteration mapping, Photogrammetric Engineering and Remote Sensing, 57 (1991) 1163-1169.
- [13] F. Douak, Reconstruction des images compressées en utilisant les réseaux de neurones artificiels et la DCT, Mémoire de magister en micro ondes pour télécommunications, Université de Batna, 2008.
- [14] F. Douak, R. Benzid, N. Benoudjit, Color image compression algorithm based on the DCT transform combined to an adaptive block scanning, AEU-International Journal of Electronics and Communications, 65 (2011) 16-26.
- [15] A. Abdelaziz, Compression d'images par fractale, Thèse de magister en informatique, Université de Batna, 2002.
- [16] A. Manzanera, TERI : Traitement et reconnaissance d'image, Cours Traitement et reconnaissance d'image, Master, Université Pierre et Marie CURIE, Paris 2006.
- [17] D. Zeroual, Implémentation d'un environnement parallèle pour la compression d'images à l'aide des fractales, Thèse de magister en informatique, Université de Batna, 2006.
- [18] O. Hammami, Etudes d'optimisation algorithmiques de JPEG2000 (EIRE), ministère de l'industrie dans le cadre du Réseau National de la Recherche en Télécommunications (RNRT), février 2004.
- [19] J. Marconi, M. Rodrigues, Transfert Sécurisé D'images par Combinaison de Techniques de Compression, Cryptage et Marquage, thèse doctorat de l'université Montpellier II Mention Informatique, 31 Octobre 2006.

- [20] T. TOTOZAFINY, Compression D'images Couleur pour Application a la Télésurveillance Routière par Transmission Vidéo à Très Bas Débit, thèse doctorat de l'université de Pau et des pays de L'Adour, Ecole Doctorale des Sciences Exactes et de Leurs Applications, 3 juillet 2007.
- [21] M. AMMAR, Optimisation D'un Schéma De Codage D'image A Base D'une TCD. Application A Un Codeur JPEG Pour L'enregistrement Numérique A Bas Débit', thèse doctorat de l'Ecole Nationale Supérieure des télécommunications, 14 Janvier 2002.
- [22] S. FASSINO, Agrandissement D'Images et de Séquences Vidéo, thèse de Institut national polytechnique de Grenoble, 13 juillet 2004.
- [23] J.M.M. Rodrigues, Transfert Sécurisé D'images par Combinaison de Techniques de Compression, Cryptage et Marquage, thèse doctorat de l'université Montpellier II Mention Informatique, 31 Octobre 2006.
- [24] E.L. PENNEC, Compression D'Image, Laboratoire de Probabilités et Modèles Aléatoires UMR 7599, Université Paris.
- [25] C. Sibade, Compression de Données pour les Systèmes de Traitement de Document Grand Format, thèse doctorat de l'Université de Marne-la-Vallée (spécialité informatique), 15 décembre 2003.
- [26] A.B. ATITALLAH, Etude et Implantation d'Algorithmes de Compression D'images dans un Environnement Mixte Matériel et Logiciel, thèse doctorat de l'université bordeaux 1, école doctorale des sciences physiques et de l'ingénieur, 11 Juillet 2007.
- [27] M. Domariski, K. Rakowski, A Simple Technique for Near-Lossless Coding of Color Images, 0-7803-5482-6/99/IEEE, International Symposium on Circuits and Systems, pp. 299-302, May 28-31 2000, Geneva, Switzerland.
- [28] I. Jolliffe, Principal component analysis, John Wiley & Sons, Ltd, 2002.
- [29] S.Y. Ooi, et al., Image-based handwritten signature verification using hybrid methods of discrete Radon transform, principal component analysis and probabilistic neural network, Applied Soft Computing, 40 (2016) 274-282.
- [30] R. Benzid, F. Marir, N.-E. Bouguechal, Electrocardiogram Compression Method Based on the Adaptive Wavelet Coefficients Quantization Combined to a Modified Two-Role Encoder, IEEE signal processing letters, 14 (2007) 373-376.
- [31] R.C. Gonzalez, R.E. Woods, S.L. Eddins, Digital Image Processing Using MATLAB, Prentice-Hall, (2004).