

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur Et de La Recherche Scientifique



Université de Ghardaïa

Faculté des Sciences et Technologies
Département d'automatique et électromécanique

N° d'ordre :
N° de série :

Mémoire présenté en vue de l'obtention du diplôme de

MASTER

Domaine : *Sciences et Technologies*

Filière : *Automatique*

Spécialité : *Automatique et Système*

Par : ▪ **GRINE Salah Eddine**
▪ **BOUASRIA Ali Aymen**

Thème

**Compression Vidéo Par La Méthode
De Transformée En Cosinus Discrète
(DCT)**

Soutenu publiquement le 19/06/2023

Devant le jury :

Mr. BITEUR Kada	MAA	Université	Président
Mr. KIFOUCHE Abdessalam	MAA	Université	Examineur
Mr. ARIF Mohammed	MAA	Université	Examineur
Mr. LADJAL Boumadiane	MAA	Université	Encadreur

Année Universitaire: 2022/2023

سُبْحَانَكَ اللَّهُمَّ رَبِّ السَّمَاوَاتِ السَّبْعِ وَالْأَرْضِ وَالْعَرْشِ الْمَجِيدِ

شُكْرُكُمْ وَتَقْدِيرُكُمْ

قال الله تعالى في كتابة الكريم: { وَمَنْ يَشْكُرْ فَإِنَّمَا يَشْكُرُ لِنَفْسِهِ }

[سورة لقمان الآية 12]

وبعد فإننا نحمد الله عز وجل حمداً كثيراً طيباً يملأ السماوات والأرض ونشكره قبل أي شيء على ما أكرمنا به من إتمام هذه الدراسة وأرجو من الله أن تنفعنا في ديننا ودنيا و ننالوا بها رضا الله .

واهتداء بهدي النبي صلى الله عليه وسلم بقوله: { مَنْ لَمْ يَشْكُرِ النَّاسَ لَمْ يَشْكُرِ اللَّهَ }
«رواه الترمذي»

فمن هنا نغتنم الفرصة لكي نتقدم بجزيل الشكر و الامتنان إلى أولياء أمورنا و فخرنا بدعمهم المادي و المعنوي لنا طيلة مشوارنا الدراسي

كما نشوجه بالشكر والامتنان لأستاذنا الفاضل الدكتور « لعجال بومديني » فقد كان لإشرافه

الفضل الكبير في انجاز هذه المذكرة متمنين له التوفيق في مساعيه المستقبلية وجزاه الله عنا كل خير

والشكر موصول لأعضاء لجنة المناقشة الكرام على تفضلهم بقبول مناقشة مذكرة التخرج هذه ولكل

أساتذة كلية العلوم و التكنولوجيا و خصوصاً أساتذة قسم الآلية والكهروميكانيك كل باسمه و كل بمقامه

شكراً لكل من ساهم في هذا العمل من قريب أو بعيد

الإهداء

قال رسول الله (صلي الله عليه وسلم):

"من صنع إليكم معروفاً فكافئوه، فإن لم تجدوا ما تكافئونه به فادعوا له حتى تروا أنكم كافأتموه

(رواه أبو داود)

إلى من ساندتني في صلاتها و دعائها إلى من تشاركني أفراحي وآساتي
. إلى من سهره الليالي تنير دربي إلى نبع العطفه و الحنان

إلى أجمل ابتسامة في حياتي، إلى أروع امرأة في الوجود : أمي الغالية خديجة

إلى من علمني أن الدنيا كفاح و سلاحها العلم و المعرفة

إلى الذي لم ييخل علي بأي شيء إلى من سعى لأجل راحتي و نجاحي

إلى أعظم وأعز رجل : أبي العزيز علي

إلى سندي و عضدي إلى إخوتي "عبد الله " " فيصل " " عبد الرحمان "

" إلى من افتخر بها إلى نعمة الله علينا أختي " سهام "

إلى الأرواح الطاهرة التي فارقتنا اسكنها الله جنة الفردوس

إلى كل من تفضل علي و لقنني من علمه



إلى من تمنوا لي الخير سراً أو جهراً، إلى كل من أرسل لي نواياه الطيبة و الى كل من ساهم في هذا العمل

.شكراً من القلب.

صلاح الدين

سجدة الإِهْتِكَاءِ

قال الله تعالى :

﴿إِنَّ اللَّهَ وَمَلَائِكَتَهُ يُصَلُّونَ عَلَى النَّبِيِّ يَا أَيُّهَا الَّذِينَ آمَنُوا صَلُّوا عَلَيْهِ وَسَلِّمُوا تَسْلِيمًا﴾

[الأحزاب 56]

"الْحَمْدُ لِلَّهِ الَّذِي بِنِعْمَتِهِ تَتِمُّ الصَّالِحَاتُ وَتُنْقَضِي الْحَاجَاتُ، وَتَأْتِي الْخَيْرَاتُ وَتَزُولُ الْعَقَبَاتُ الْحَمْدُ لِلَّهِ الَّذِي
وفقني وأهمني الصبر والسلوان حتى وصلت إلى ما أنا عليه الآن وأكمل مسيرتي بالتخرج

أهدي تخرجي لكل من :

إلى أجمل هدية من الله و أرحم نفس و أطيب روح التي سهرت الليالي الطوال متمنيتا و داعيتا لفلاحي و

نجاحي أُمِّي الغالية رحيمة بارك الله فيها و أطال عمرها

إلى من علمني تخطي الصعاب و أنشأني و حرص على توجيهي و دعمي كل الدعم أبي الغالي سليمان أدامه الله

تاج فوق رؤوسنا

إلى الأرواح الطاهرة التي لقيناها و من لم نلقاها تغمدها الله برحمته

إلى سندي و عضدي أخي محمد رضوان إلى أخواني دعاء و سيدرة أنار الله درهم و سدّد خطاهم

اهدي لكم هاته الكلمات التي لا تكاد تصف حيي و امتناني لكم

إلى أستاذي الكريم الذي بذل كل الجهد و لم ييخل علينا بحرف

إلى رفيق الدرب صلاح الدين جعله الله صالحا فالح و الصديق العزيز عماد الدين وفقه الله و الأخ الكبير يونس

الذي كان معلما و موجه و إلى كل من جمعنا بهم أروقة العلم شكرا لكم كل باسمه و كل بمقامه

شكرا خالصا من القلب

علي أيمن

Résumé:

Les vidéos numériques rencontrent des problèmes de traitement, de stockage et de transmission en raison de leur grande taille. Le but de la compression est de réduire la quantité de données que nous devons stocker ou transmettre sous forme de vidéo, images, audio, etc. Il existe une gamme d'algorithmes de compression qui répondent aux nouvelles applications.

Ce mémoire vise à explorer la méthode de compression par la DCT pour les vidéos surtout les vidéos de surveillance. En comprenant les fondements de la compression d'images et de vidéos, ainsi que les principes et les techniques spécifiques de la compression par la DCT, nous pourrions évaluer son efficacité et sa pertinence dans différents scénarios. Les résultats et les conclusions tirés de cette étude, qui a été menée à l'aide du logiciel MATLAB pourront servir de référence dans le domaine de la compression vidéo et contribuer à l'amélioration des méthodes existantes.

Mots clés: Transformée en cosinus discrète ; TCD ; Compression des données; Compression d'une vidéo ; Traitement d'images et vidéos ; algorithmes .

Abstract:

Digital videos encounter processing, storage and transmission problems due to their large size. The purpose of compression is to reduce the amount of data we have to store or transmit in the form of video, images, audio, etc. There is a range of compression algorithms that cater to new applications.

This thesis aims to explore the DCT compression method for videos, especially surveillance videos. By understanding the fundamentals of image and video compression, as well as the specific principles and techniques of DCT compression, we will be able to assess its effectiveness and suitability in different scenarios. The results and conclusions drawn from this study, which was conducted using MATLAB software, may serve as a reference in the field of video compression and contribute to the improvement of existing methods.

Keywords: Discrete cosine transform; DCT ; data compression; Video compression; Image and video processing; algorithms.

الملخص:

تواجه مقاطع الفيديو الرقمية مشاكل في المعالجة والتخزين والنقل بسبب حجمها الكبير. الغرض من الضغط هو تقليل كمية البيانات التي يتعين علينا تخزينها أو نقلها في شكل فيديو أو صور أو صوت وما إلى ذلك. هناك مجموعة من خوارزميات الضغط التي تناسب التطبيقات الجديدة.

تهدف هذه المذكرة إلى استكشاف طريقة ضغط DCT لمقاطع الفيديو خاصة فيديوهات المراقبة. من خلال فهم أساسيات ضغط الصور والفيديو، بالإضافة إلى المبادئ والتقنيات المحددة لضغط DCT، سنكون قادرين على تقييم فعاليتها ومدى ملاءمتها في سيناريوهات مختلفة. يمكن أن تكون النتائج والاستنتاجات المستخلصة من هذه الدراسة التي أجريت باستخدام برنامج MATLAB بمثابة مرجع في مجال ضغط الفيديو والمساهمة في تحسين الأساليب الحالية.

الكلمات المفتاحية: تحويل جيب التمام المنفصل DCT؛ ضغط البيانات؛ ضغط الفيديو؛ معالجة الصور والفيديو؛ الخوارزميات.

LISTE DES FIGURES

LISTE DES FIGURES

Figure I.1 : Représentation de pixel.....	5
Figure I.2: Image sans bruit.....	6
Figure I.3: Image avec bruit.....	6
Figure I.4 : Contour d'une image.....	7
Figure I.5 : L'histogramme de l'image.....	7
Figure I.6 : Image binaire.....	8
Figure I.7 : Image à niveaux de gris.....	8
Figure I.8 : Image couleurs.....	9
Figure I.9 : Différence entre l'image vectorielle et l'image matricielle.....	9
Figure I.10 : Types de traitements d'images et vidéos.....	17
Figure II.1 : Schéma de compression /décompression classique d'image.....	29
Figure II.2 : Exemple de technique intra-image par transformation des couleurs.....	35
Figure II.3: Exemple de technique intra-image par Elimination les redondances temporelle.....	35
Figure II.4 : Technique intra-image (compression spatiale).....	36
Figure II. 5 : Exemple sur la technique Inter-image par le codage différentiel.....	36
Figure II.6 : Comparer les blocs.....	37
Figure II.7 : Exemple sur la technique Inter-image par La compensation de mouvement...	37
Figure II.8 : La classification des images en différents types (I, P ou B).....	38
Figure II.9 : Modules d'un système de compression vidéo.....	38
Figure III.1: Architecture de Compression et la décompression.....	44
Figure III.2: Exemple de zigzag sur la table de quantification.....	48

LISTE DES TABLEAUX

LISTE DES TABLEAUX

Le Tableau III.1 : Exemple de diviser une vidéo en images de taille 8x8.....	46
Le Tableau III.2 : Exemple de Coefficients de fréquence.....	46
Le Tableau III.3 : Exemple de Table de quantification.....	47
Le Tableau III.4 : Exemple de Coefficients quantifiés.....	48
Tableau III.5 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.....	52
Tableau III.6 : représente Paramètres d'évaluation de test 1.....	52
Tableau III.7 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.....	53
Tableau III.8 :représente Paramètres d'évaluation de test 2.....	53
Tableau III.9 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.....	54
Tableau III.10 :représente Paramètres d'évaluation de test 3.....	54

LISTE DES ABREVIATIONS

DCT : Discret Cosine Transform.

TCD : Transformée en Cosinus Discrète .

TFD : Transformée de Fourier Discrète.

TOC : Transformation en ondelettes continue.

TOD : Transformation en ondelettes discrètes.

DFT: Discret Fourier Transform.

Bpp : Bit Par Pixel.

PPP : Points Par Pouce.

DPI: Dots Per Inch.

BPS : Bits Par Seconde.

MBPS : Mégabits Par Seconde .

RLE: Run Length Encoding.

AVI: Audio Video Interleave.

CCD :Charged Coupled Device.

Codec :Encodeur/Décodeur.

DVD: Digital Vertical Disk.

Flv :Flash Video.

SECAM :Séquentiel Couleur A Mémoire.

NTSC: National Television System Commette.

PAL: Phase Alternation Line.

MKV: Matroska Video.

Mov: Quicktime Movie.

MP4: MPEG-4Part 14.

WMV :Windows Media Video .

VHS :Vidéo Home System.

HD : Haute Définition.

LISTE DES ABREVIATIONS

UHD : Ultra Haute Définition.

HSL: Hue Saturation Lightness.

CMJN : Cyan Magenta Jaune Noir.

RGB: Red Green Blue.

RVB: Rouge Vert Bleu.

YCbCr : Luminance (Y), Chrominance (Red-Yellow).

ISO: International Standards Organisation.

JPEG: Joint Photographic Experts Group.

GIF : Graphic Interchange Format.

PNG: Portable Network Graphic.

JPEG : 2000: Joint Photographic Experts Group 2000.

MPEG: Moving Picture Expert Group.

AVC: Advanced Video Coding.

HEVC: High Efficiency Video Coding.

CR : Le Taux de Compression.

GC : Le Gain de Compression.

MSE: Mean Square Error.

SNR: Signal to Noise Ratio.

PSNR : Peak Signal to Noise Ratio.

SOMMAIRE

Sommaire

Remerciement

Dédicaces

Résumé

Listes des figures

Liste des tableaux

Liste des abréviations.

Introduction générale 1

Chapitre I : Traitement D'images Et Vidéos

I.1 Introduction : 4

I.2 Manipulation des images et vidéos : 4

I.3 L'images : 4

I.3.1 Définition d'une image : 4

I.3.2 Image numérique : 4

I.3.3 Caractéristiques d'une image numérique : 5

I.3.4 Type d'image : 8

I.3.5 Images Bitmap et image vectorielles : 9

I.4 la vidéo : 10

I.4.1 Définition : 10

I.4.2 Les objets : 10

I.4.3 Les paramètres clés d'une vidéo : 10

I.4.4 Types de vidéo : 11

I.4.5 Les format vidéos : 11

I.4.6 Les standards : 13

I.4.7 Codages : 14

I.4.8 Le montage vidéo : 14

I.5 Domaine d'application (Types de traitements d'images et vidéos) : 16

I.6 Conclusion : 18

Chapitre II : Compression d'images et vidéos : Techniques et normes

II.1 Introduction : 20

II.2 Définition de la compression : 20

II.3 Les objectifs de la compression : 20

II.4 la compression des données: 21

SOMMAIRE

II.4.1 Types de Compression:	21
II.6 Compression d'image numérique :	27
II.6.1 Le principe général de la compression d'images	27
II.6.2 Définition des critères utilisés en compression :	29
II.6.3 La décompression d'image :	30
II.6.4 Les norme de compressions d'image:	31
II.7 Compression de la vidéo numérique:	31
II.7.1 Codec:	32
II.7.2 Technique de compression Vidéo :	32
II.7.3 Les normes de compression d'une vidéo :	37
II.9 Conclusion :	38

Chapitre III : Compression Vidéo Par DCT

III.1 Introduction :	40
III.2 Définition de Matlab :	40
III.3 Transformation en cosinus discrète :	41
III.3.1 Variantes de la DCT :	41
III.4 Architecture de Compression et la décompression DCT :	42
III.4.1 La compression et la décompression par la DCT :	43
III.5 Tests et Résultats :	50
III.5.1 Logiciel de l'implémentation :	50
III.5.2 Paramètres d'évaluation :	50
III.5.3 Test 1:	51
III.5.4 Test 2 :	52
III.5.5 Test 3 :	53
III.5.6 Interprétations des tests :	54
III.6 Conclusion :	55
Conclusion générale :	57

Bibliographie

Introduction
Générale

Introduction générale

Introduction générale

Dans le monde numérique axé sur le partage d'informations, la vitesse et la capacité de transmission jouent un rôle essentiel. C'est pourquoi la compression des données est devenue de plus en plus importante. Son objectif est de réduire le nombre de bits nécessaires pour transmettre ou stocker les informations. Les techniques de compression/décompression ont été développées dans le but d'économiser les ressources de stockage et d'optimiser l'utilisation des canaux de communication. Les algorithmes utilisés dans ces techniques, qu'il s'agisse de compression d'images, de vidéos ou autres, reposent sur diverses approches comme le traitement du signal, le traitement d'images et le codage d'information. Ces techniques produisent des signaux codés qui malheureusement peuvent être sensibles aux erreurs de transmission, ce qui peut affecter la qualité des informations décodées. [1]

Le premier système de compression de données, connu sous le nom de code "Morse", a été inventé par Samuel Morse en 1836. Aujourd'hui, nous avons besoin de la compression de données pour plusieurs raisons. L'une d'entre elles est de libérer de l'espace de stockage occupé par des fichiers volumineux, ce qui nous permet de stocker davantage de données sur le même espace de stockage. En ce qui concerne les réseaux de communication, la compression des données revêt une importance capitale. Avec l'augmentation des fichiers volumineux tels que les fichiers audio, les images et autres, l'envoi de ces fichiers sans compression nécessiterait beaucoup de temps pour les processus de transmission et de réception, entraînant un engorgement du trafic réseau. En compressant les mêmes données avant de les transférer, elles se déplacent beaucoup plus rapidement, réduisant ainsi le temps de transmission.[1]

La compression des données, que ce soit pour les vidéos ou autres types de fichiers, consiste essentiellement à éliminer les informations superflues, réduisant ainsi la redondance. La redondance se réfère à la présence excessive de données par rapport à la logique. Elle peut avoir une connotation négative de superflu, mais peut également être perçue positivement lorsque cette redondance est intentionnelle afin de prévenir les dysfonctionnements. Dans le cadre de ce mémoire, nous nous intéressons à l'une des méthodes de compression irréversible, à savoir la compression des vidéos à l'aide de la Transformée en cosinus discrète (TCD).[2]

La Transformée en cosinus discrète (TCD) est une technique de transformation similaire à la Transformée de Fourier discrète (TFD), mais avec la particularité que son noyau de

Introduction générale

projection est un cosinus, ce qui génère des coefficients réels. Contrairement à la TFD, qui utilise des exponentielles complexes comme noyau et produit donc des coefficients complexes. La TCD peut cependant être exprimée en fonction de la TFD, qui est appliquée sur un signal symétrique. [3]

En utilisant la TCD, les séquences d'images peuvent être divisées en blocs plus petits, puis transformées en coefficients de fréquence. Cette transformation mathématique permet de réduire la taille des fichiers vidéo, tout en préservant une qualité visuelle acceptable.

Dans ce mémoire il y'a trois chapitres, selon l'organisation suivante :

Le premier chapitre est consacré à la présentation des différents outils permettant le traitement des images et vidéos numérique et domaine d'application

Ensuite, nous nous concentrerons sur la compression des données, en fournissant une définition de la compression et en exposant les objectifs de ce processus. Nous discuterons des différents types de compression, y compris la compression sans perte et la compression avec perte, ainsi que des techniques et des algorithmes de compression utilisés dans images et vidéo numérique

Dans le dernier chapitre, nous aborderons spécifiquement la transformation en cosinus discrète (DCT) comme méthode de compression. Nous expliquerons les principes de la DCT et son utilisation dans la compression de vidéos (les vidéos de surveillance.) Nous présenterons également une architecture de compression et de décompression basée sur la DCT, ainsi que des tests et des résultats pour évaluer son efficacité. Ce travail est achevé par une conclusion générale

Chapitre

I

I.1 Introduction :

Le traitement d'images et vidéos est un domaine de l'informatique qui s'est considérablement développé ces dernières années. Les images et vidéos sont omniprésentes dans notre vie quotidienne, que ce soit dans les médias, les réseaux sociaux, les publicités ou encore les applications mobiles.

Dans ce chapitre, nous allons explorer les différentes techniques de manipulation d'images et vidéos, en commençant par les fondements de l'image numérique, ses caractéristiques et ses différents types. Nous allons également examiner les concepts clés liés à la vidéo, tels que les formats, les standards et les codages. Enfin, nous allons explorer les différents types de traitements d'images et vidéos et leurs domaines d'application.

I.2 Manipulation des images et vidéos :

La manipulation des images et vidéos consiste à effectuer des opérations sur ces dernières pour les modifier, les améliorer ou les analyser. Ces opérations peuvent inclure la correction des couleurs, la réduction du bruit, l'amélioration de la netteté, la compression, la conversion entre différents formats d'image ou de vidéo, et bien plus encore.

I.3 L'images :

I.3.1 Définition d'une image :

Une image est une représentation visuelle d'un objet ou d'une scène. Elle peut être capturée par un dispositif optique tel qu'un appareil photo ou une caméra, ou créée à l'aide d'un logiciel de dessin. Une image peut être en couleur ou en noir et blanc, C'est aussi un ensemble structuré d'informations qui, après affichage sur l'écran, a une signification pour l'œil humain et elle peut être représentée sous forme analogique ou numérique [4]

I.3.2 Image numérique :

Une image numérique est une image qui est stockée sous forme des données numériques (série de bits) dans un ordinateur. Elle peut être obtenue en numérisant une image analogique à l'aide d'un scanner, ou en capturant une image à l'aide d'un appareil photo numérique ou d'une caméra. aussi par intelligence artificielle . L'image numérique est l'image dont la surface est divisée en éléments de taille fixe appelés pixels et représentée par une matrice bidimensionnelles des valeurs numériques

f(i, j) où :

i, j : coordonnées cartésiennes d'un point de l'image.

$f(i, j)$: niveau de gris en ce point.[5]

I.3.3 Caractéristiques d'une image numérique :

I.3.3.1 pixel :

Pixel est le terme anglais utilisé pour désigner un élément d'image numérique. C'est la plus petite unité qui compose une image et peut être considérée comme une brique de construction individuelle de l'image. Les pixels sont utilisés dans diverses technologies numériques telles que les caméras numériques, les écrans d'ordinateur, les téléviseurs, les smartphones et les consoles de jeux vidéo



Figure I.1 : représentation de pixel

I.3.3.2 La résolution :

La résolution fait référence à la quantité de détails que l'on peut distinguer dans une image, qu'elle soit affichée sur un écran ou imprimée. La résolution est mesurée en points par pouce (ppp) ou dpi (dots per inch).[6]

I.3.3.3 Dimension :

Les dimensions d'une image numérique sont déterminées par le nombre de pixels qui composent l'image. Les dimensions sont généralement exprimées en largeur et en hauteur en pixel. Par exemple, une image de 1920 pixels de largeur et 1080 pixels de hauteur a une dimension de 1920x1080 pixels.[7]

I.3.3.4 Le bruit :

Le bruit est un phénomène qui se produit lorsque des perturbations aléatoires sont présentes dans l'image, créant ainsi des pixels indésirables qui peuvent affecter la qualité de

l'image. Le bruit peut être dû à plusieurs facteurs, tels que des conditions d'éclairage faibles, une sensibilité ISO élevée, une compression excessive ou un mauvais traitement de l'image. [8]



Figure I-2: Image sans bruit **Figure I-3:** Image avec bruit

I.3.3.5 La luminance :

La luminance correspond au niveau de luminosité des points de l'image. Elle est définie comme étant le rapport entre l'intensité lumineuse d'une surface et son aire apparente. Pour un observateur éloigné, la luminance remplace la notion de brillance qui décrit l'éclat d'un objet. Une bonne luminance se caractérise par :

- Des images luminousness (brillantes).
- Un bon contraste : il faut éviter les images où la gamme de contraste tend vers le blanc ou le noir, car cela entraîne des pertes de détails dans les zones sombres ou lumineuses.
- L'absence de parasites. [9]

I.3.3.6 Les contours et Textures :

Les contours d'une image représentent la limite entre les objets de l'image ou la frontière entre deux pixels ayant des niveaux de gris significativement différents. Les textures, quant à elles, décrivent la structure de ces objets. L'extraction de contours consiste à identifier les points de l'image qui séparent deux textures différentes [10]

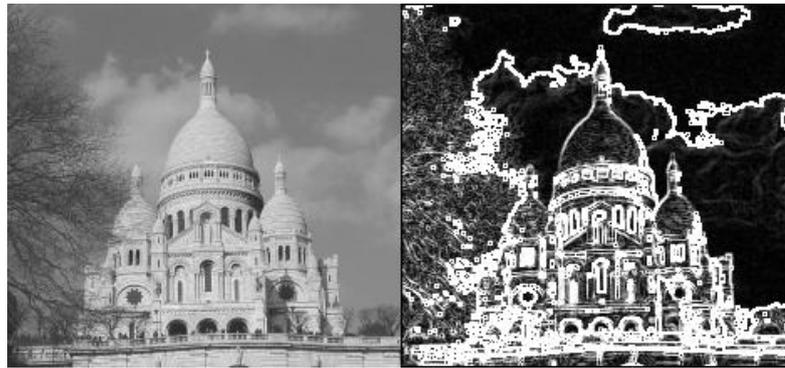


Figure I.4 : Contour d'une image

I.3.3.7 L'histogramme :

L'histogramme d'une image, qu'il soit en niveaux de gris ou en couleurs, représente la fréquence d'apparition de chaque niveau de gris (couleur) dans l'image. Il permet de réduire l'erreur de quantification, de comparer deux images prises sous des éclairages différents ou encore de mesurer certaines propriétés sur une image. L'histogramme fournit également de nombreuses informations sur la distribution des niveaux de gris (couleur) et permet de déterminer la plage de valeurs dans laquelle la majorité des niveaux de gris (couleur) est répartie, dans les cas où l'image est trop claire ou trop foncée[11].

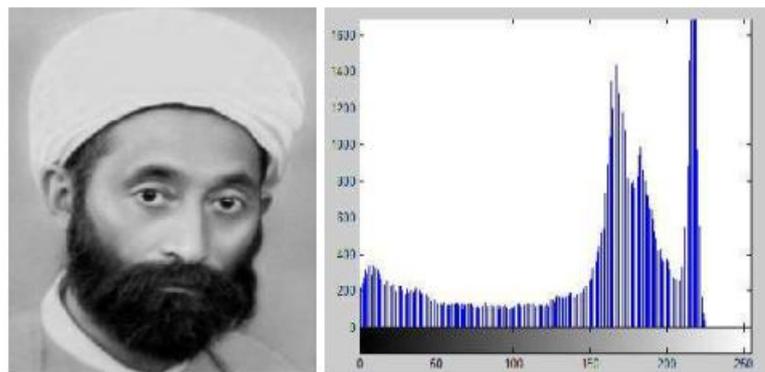


Figure I.5: L'histogramme de l'image

I.3.3.8 Le 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. Si L_1 et L_2 sont les degrés de luminosité respectivement de deux zones voisines A_1 et A_2 d'une image, le contraste C est défini par le rapport :[9]

$$C = \frac{L_1 - L_2}{L_1 + L_2} \quad (\text{I.1})$$

I.3.4 Type d'image :

I.3.4.1 Images binaires (en noir et blanc):

Les images binaires sont constituées uniquement des pixels noirs et blancs, sans nuances de gris ou de couleur. Chaque pixel est codé en un seul bit, où 0 représente le noir et 1 représente le blanc.



Figure I.6 :Image binaire

I.3.4.2 Images à niveaux de gris (Monochromes) :

Les images à niveaux de gris ont des pixels qui varient en nuances de gris, allant du blanc au noir. Chaque pixel est codé sur plusieurs bits, généralement de 8 à 16 bits, pour permettre une plus grande gamme de nuances de gris.



Figure I.7 :Image à niveaux de gris

I.3.4.3 Images en couleurs (Polychromes):

Les images en couleurs ont des pixels qui représentent des couleurs différentes. Chaque pixel est codé sur plusieurs bits pour permettre une grande variété de couleurs, généralement 24 bits pour les images RVB (rouge, vert, bleu). Les images en couleurs peuvent également

être représentées dans d'autres espaces de couleurs, tels que CMJN (cyan, magenta, jaune, noir).



Figure I.8 :Image couleurs.

I.3.5 Images Bitmap et image vectorielles :

Les images bitmap ou raster sont composées d'une matrice de pixels, chaque pixel contenant des informations de couleur et de luminosité. En raison de leur résolution fixe, ces images peuvent paraître floues ou pixelisées lorsqu'elles sont agrandies ou réduites, car chaque pixel est agrandi ou réduit en taille, entraînant une perte de qualité globale.

Les images vectorielles, quant à elles, sont constituées de formes géométriques, de lignes et de courbes définies par des équations mathématiques. Elles peuvent être redimensionnées sans perte de qualité, car les formes géométriques sont recalculées en fonction de la nouvelle taille. [12]



Figure I.9 : Différence entre l'image vectorielle et l'image matricielle

I.4 la vidéo :

I.4.1 Définition :

La vidéo est une technologie qui permet d'enregistrer et de stocker séquentiellement des images animées avec ou sans sons sur un support numérique, et de les afficher sur un écran d'affichage de manière séquentielle. Une vidéo est constituée d'une série d'images successives, et ces images sont lues si rapidement que l'impression de mouvement est créée lors de l'affichage de la vidéo.

I.4.2 Les objets :

Dans le domaine de la vidéo, les objets sont des éléments en mouvement qui sont enregistrés sur une séquence vidéo. Selon la définition du standard MPEG-7, un objet vidéo est une "région spatiale de l'image vidéo qui est distinguée par des caractéristiques visuelles ou sémantiques et qui est significative pour la recherche, l'analyse ou la description".

I.4.3 Les paramètres clés d'une vidéo :

I.4.3.1 Les frames :

Les frames ou images dans une vidéo sont des images fixes qui sont affichées à une fréquence suffisamment rapide pour créer l'illusion de mouvement continu. Le nombre de frames par seconde est un paramètre important qui influe sur la qualité et la fluidité de la vidéo. Une vidéo de 30 fps est considérée comme étant fluide et convient pour la plupart des utilisations. [13]

I.4.3.2 La résolution :

La résolution fait référence au nombre de pixels dans l'image vidéo. Plus la résolution est élevée, plus l'image est nette et détaillée. Les résolutions courantes pour la vidéo numérique comprennent 720p, 1080p, 4K et 8K.

I.4.3.3 Le débit binaire :

Le débit binaire, ou bitrate en anglais, est la quantité de données utilisée pour stocker une seconde de vidéo. Plus le débit binaire est élevé, plus la qualité de la vidéo est élevée, mais plus le fichier vidéo est lourd. Le débit binaire est mesuré en bits par seconde (bps) ou en mégabits par seconde (Mbps).

I.4.3.4 Le ratio d'aspect :

Le ratio d'aspect d'une vidéo se réfère à la proportion entre la largeur et la hauteur de l'image. Il est exprimé sous forme de deux nombres séparés par deux points, comme 16:9 ou 4:3. Le choix du ratio d'aspect dépend de l'utilisation prévue de la vidéo, de l'appareil sur lequel elle sera visionnée, et des préférences esthétiques du créateur.

I.4.3.5 Le format :

Le format d'un fichier vidéo détermine la façon dont les données de la vidéo sont organisées et stockées. Il peut également influencer la qualité vidéo et la compatibilité avec les différents dispositifs.

I.4.4 Types de vidéo :

On distingue deux grandes familles de systèmes vidéo : les systèmes vidéo analogiques et les systèmes vidéo numériques.

I.4.4.1 Les vidéos analogiques :

Les vidéos analogiques sont des signaux électriques continus qui sont modulés pour transporter des informations vidéo. Les signaux analogiques peuvent être de différentes résolutions, mais la qualité de l'image est souvent limitée par le bruit et la distorsion.

I.4.4.2 Les vidéos numériques :

Les vidéos numériques, en revanche, sont constituées de séquences de 0 et de 1, appelées bits. Ces bits sont regroupés en octets et en paquets pour former des données vidéo. Les vidéos numériques offrent une qualité d'image supérieure à celle des vidéos analogiques, car elles sont moins sensibles au bruit et à la distorsion.

I.4.5 Les formats vidéos :**I.4.5.1 Numériques :**

- **MP4** : Il s'agit d'un format de fichier vidéo couramment utilisé pour les vidéos en ligne et les appareils mobiles. Il utilise une compression efficace pour réduire la taille du fichier tout en conservant une qualité vidéo élevée.
- **MOV** : Il s'agit d'un format de fichier vidéo développé par Apple et utilisé principalement pour les vidéos QuickTime. Il prend en charge plusieurs pistes vidéo et audio ainsi que des métadonnées.
- **WMV** : Il s'agit d'un format de fichier vidéo

développé par Microsoft et utilisé principalement pour les vidéos Windows Media Player. Il prend en charge la compression efficace des fichiers vidéo tout en conservant une qualité élevée.

- **AVI** : Il s'agit d'un format de fichier vidéo populaire utilisé pour les vidéos de haute qualité. Il prend en charge plusieurs pistes audio et vidéo ainsi que des métadonnées.
- **AVCHD** : Il s'agit d'un format de fichier vidéo haute définition utilisé principalement pour les caméscopes numériques. Il utilise une compression efficace pour réduire la taille du fichier tout en conservant une qualité vidéo élevée.
- **FLV, F4V et SWF** : Ce sont des formats de fichiers vidéo développés par Adobe pour une utilisation en ligne. Ils sont principalement utilisés pour les vidéos en streaming et les animations web.
- **MKV** : Il s'agit d'un format de fichier vidéo open source qui prend en charge plusieurs pistes audio et vidéo ainsi que des sous-titres. Il offre une qualité vidéo élevée et prend en charge la compression efficace des fichiers.
- **WEBM ou HTML5** : Il s'agit d'un format de fichier vidéo open source utilisé pour les vidéos en ligne. Il offre une qualité vidéo élevée et utilise une compression efficace pour réduire la taille du fichier. [14]

I.4.5.2 Analogiques :

- **VHS (Vidéo Home System)** : un format de cassette magnétique développé par JVC et introduit en 1976. Il a été largement utilisé pour l'enregistrement de vidéos domestiques jusqu'à l'arrivée des DVD.
- **Betamax** : un format de cassette magnétique développé par Sony et introduit en 1975. Bien que la qualité de l'image soit supérieure à celle de VHS, il a été supplanté par le format VHS en raison de la durée d'enregistrement plus longue de VHS.
- **Video8 (ou Hi8)** : un format de cassette magnétique développé par Sony et introduit en 1985. Il était principalement utilisé pour les caméscopes grand public.
- **Umatic** : un format de cassette magnétique développé par Sony et introduit en 1969. Il était principalement utilisé pour la production professionnelle et les émissions de télévision.
- **Betacam** : un format de cassette magnétique développé par Sony et introduit en 1982. Il était principalement utilisé pour la production professionnelle et les émissions de télévision de qualité supérieure.

- **D-VHS** : un format de cassette magnétique développé par JVC et introduit en 1998. Il a offert une qualité d'image supérieure et une capacité de stockage plus élevée que les formats VHS et Betamax, mais n'a jamais atteint une large adoption.

I.4.6 Les standards :

Les normes vidéo sont des spécifications techniques qui définissent les caractéristiques des signaux vidéo, tels que la résolution, le format d'image, le taux de rafraîchissement et les normes de codage. Ces normes permettent de garantir la compatibilité entre les différents équipements vidéo, tels que les caméras, les moniteurs, les téléviseurs et les lecteurs multimédias.

Les normes vidéo les plus courantes sont:

- **NTSC (National Television System Committee)** : c'est le standard de vidéo analogique utilisé principalement en Amérique du Nord, en Amérique centrale et dans certains pays d'Amérique du Sud. La résolution standard est de 720 x 480 pixels à 30 images par seconde.
- **PAL (Phase Alternating Line)** : c'est le standard de vidéo analogique utilisé principalement en Europe, en Afrique, en Asie et en Australie. La résolution standard est de 720 x 576 pixels à 25 images par seconde.
- **SECAM (Séquentiel Couleur avec Mémoire)**: c'est le standard de vidéo analogique utilisé principalement en France, en Russie, en Afrique du Nord et dans certains pays d'Asie. La résolution standard est de 720 x 576 pixels à 25 images par seconde.

Ces normes sont utilisées pour les signaux analogiques. Cependant, avec l'avènement de la technologie numérique, de nouvelles normes ont été développées, telles que:

- **HD (Haute Définition)**: avec une résolution de 1280 x 720 pixels ou 1920 x 1080 pixels.
- **UHD (Ultra Haute Définition)**: avec une résolution de 3840 x 2160 pixels.
- **4K**: avec une résolution de 4096 x 2160 pixels.
- **8K**: avec une résolution de 7680 x 4320 pixels.

Ces normes vidéo numériques utilisent différentes méthodes de compression, telles que MPEG-2, MPEG-4 et H.264, pour réduire la taille des fichiers vidéo et permettre une transmission et un stockage plus efficaces. [14]

I.4.7 Codages :

Les codages sont des systèmes de représentation des couleurs utilisés pour afficher les vidéos. Les principaux codages sont les suivants : [15]

- **CIE / Lab.** :Le codage CIE / Lab. est un système de représentation des couleurs basé sur les propriétés physiques de la lumière. Il permet de représenter toutes les couleurs visibles par l'œil humain.
- **RGB (RVB)** :Le codage RGB (Red, Green, Blue) est le codage le plus couramment utilisé pour les écrans d'ordinateur et les téléviseurs. Il utilise des combinaisons de rouge, de vert et de bleu pour représenter toutes les couleurs possibles.
- **HSL (TSL)** :Le codage HSL (Hue, Saturation, Lightness) est un système de représentation des couleurs qui utilise la teinte, la saturation et la luminosité pour décrire une couleur.
- **CMYK (CMJN)** :Le codage CMYK (Cyan, Magenta, Yellow, Key) est principalement utilisé pour l'impression. Il utilise des combinaisons de cyan, de magenta, de jaune et de noir pour représenter toutes les couleurs possibles.

I.4.8 Le montage vidéo :

Le montage de vidéo est un processus créatif et technique qui consiste à sélectionner, trier et assembler des séquences vidéo pour créer une histoire cohérente et convaincante. Il existe plusieurs types de montage, notamment le montage linéaire traditionnel et le montage non linéaire ou "virtuel". Voici un aperçu des différents aspects du montage de vidéo : [16]

I.4.8.1 Exploitation :

L'exploitation est la première étape du montage de vidéo. Cela implique la collecte et l'organisation de toutes les séquences vidéo, les images fixes, la musique et les effets sonores nécessaires pour créer la vidéo finale.

I.4.8.2 Le banc de montage :

Le banc de montage était autrefois un outil physique utilisé pour éditer les films, mais aujourd'hui il a été remplacé par des logiciels de montage vidéo.

I.4.8.3 Le montage linéaire :

Le montage linéaire est une méthode de montage qui implique l'assemblage des séquences vidéo dans un ordre chronologique sur une bande de montage. Cette méthode est

moins flexible que le montage non linéaire mais elle est toujours utilisée dans certaines productions professionnelles.

I.4.8.4 Le montage non linéaire :

Le montage non linéaire ou « virtuel » Le montage non linéaire est une méthode de montage qui permet de travailler de manière flexible avec les séquences vidéo. Les séquences peuvent être réorganisées, modifiées, supprimées ou ajoutées facilement. Cette méthode est très populaire et est utilisée par la plupart des professionnels du montage vidéo.

I.4.8.4.1 L'acquisition :

L'acquisition est le processus d'enregistrement des séquences vidéo brutes. Cela peut se faire avec une caméra vidéo, un smartphone ou une autre source d'enregistrement.

I.4.8.4.2 Les Avantages du montage non linéaire

Le montage non linéaire offre plusieurs avantages, notamment la flexibilité pour réorganiser les séquences, la possibilité de visualiser instantanément les modifications apportées, ainsi que la possibilité d'utiliser des effets spéciaux et des transitions de manière plus créative.

I.4.8.4.3 Les Inconvénients du montage non linéaire

Les inconvénients du montage non linéaire incluent la nécessité d'un ordinateur puissant et de logiciels de montage vidéo, ainsi que le temps nécessaire pour apprendre les compétences techniques nécessaires pour utiliser les outils de montage.

I.4.8.4.4 Le mixage :

Le mixage implique la combinaison de plusieurs pistes audio pour créer une bande sonore finale. Cela peut inclure la musique, les effets sonores et les enregistrements de voix.

I.4.8.4.5 Le codage (conversion) :

Le codage ou la conversion consiste à transformer le fichier vidéo final en un format compatible avec les appareils sur lesquels la vidéo sera visionnée, tels que les ordinateurs, les téléviseurs ou les smartphones.

I.4.8.4.6 La gravure et la lecture :

La gravure consiste à enregistrer la vidéo finale sur un disque optique, tel qu'un DVD ou un Blu-ray, pour une lecture sur des appareils compatibles. La lecture peut également se faire sur des plates-formes de streaming en ligne telles que YouTube, Vimeo ou Netflix.

I.4.8.4.7 Logiciels de montage non linéaire :

Il existe de nombreux logiciels de montage non linéaire disponibles, tels que AdobePremiere Pro, Final Cut Pro, Avid Media Composer et DaVinciResolve, qui sont utilisés par les professionnels du montage vidéo pour créer des productions de haute qualité.

I.5 Domaine d'application(Types de traitements d'images et vidéos) :

Chaque jour, nous sommes confrontés à une multitude d'images dans notre environnement : des photographies de paysages et de personnes, des peintures, des dessins générés par ordinateur, des images de radiologie médicale, des images prises par des satellites, et bien d'autres encore. Certaines de ces images (par exemple, les images satellites ou médicales) ne peuvent être observées directement, tandis que d'autres présentent des caractéristiques qui doivent être extraites automatiquement, stockées et transmises. Les traitements possibles pour ces images sont très variés, car les images que nous rencontrons dans notre environnement sont très diverses, que ce soit en termes de leur nature, de leurs caractéristiques ou de la scène qu'elles décrivent. Étant donné qu'il est impossible de créer un traitement spécifique pour chaque image, il a été nécessaire de créer une classification en fonction de l'objectif du traitement, plutôt qu'en fonction des caractéristiques des images elles-mêmes. Ainsi, on peut distinguer quatre domaines d'application principaux pour le traitement d'images numériques :

- la restauration et l'amélioration d'images.
- l'analyse d'images.
- le codage avec compression d'information.
- la synthèse d'images. [17]

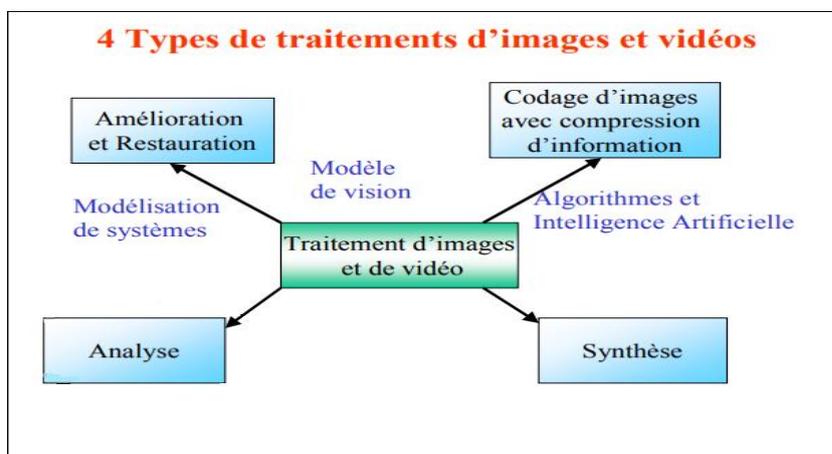


Figure I.10: Types de traitements d'images et vidéos

I.5.1 La restauration et l'amélioration d'images :

L'amélioration et la restauration d'images sont des processus importants pour préserver et améliorer la qualité visuelle des images. L'amélioration d'images implique l'application de techniques pour améliorer la qualité visuelle d'une image existante, tandis que la restauration d'images implique la récupération d'une image à partir de données endommagées ou dégradées.

Les techniques d'amélioration d'images incluent l'ajustement de la luminosité, du contraste et de la netteté, la correction des couleurs, la réduction du bruit et l'élimination des artefacts indésirables. Ces techniques peuvent être appliquées manuellement ou à l'aide de logiciels d'édition d'images.

La restauration d'images est plus complexe et nécessite souvent l'utilisation de techniques de traitement d'images avancées telles que la restauration de données manquantes ou endommagées, la suppression de rayures, de taches ou de dégradations similaires, la reconstruction de contours et de détails manquants, etc. Les algorithmes de restauration d'images sont souvent basés sur des techniques de vision par ordinateur, de traitement du signal ou d'apprentissage automatique. [17]

I.5.2 L'analyse d'images :

L'analyse d'images est une technique de traitement d'images qui consiste à extraire des informations et des caractéristiques à partir d'une image numérique. Cela implique l'utilisation d'algorithmes informatiques pour détecter, mesurer et quantifier des propriétés telles que la couleur, la texture, la forme, la taille et l'orientation des objets dans une image. L'analyse d'images est utilisée dans de nombreux domaines tels que la surveillance de la circulation, la surveillance de la sécurité, la médecine, la biologie, la forensique, la recherche scientifique et l'industrie pour automatiser des processus de traitement d'images qui seraient autrement difficiles ou impossibles à réaliser manuellement.

I.5.3 La synthèse d'images :

La synthèse d'images consiste à reconstruire une image qui ressemble à la scène simulée, en utilisant une description de la scène, des objets qui la composent, des caractéristiques d'éclairage telles que l'incidence et l'intensité, ainsi que le dispositif de prise de vue tel qu'une caméra CCD ou CMOS. Cette image synthétisée peut être une représentation de la réalité ou une création purement imaginaire. Les premières applications liées aux images de synthèse étaient orientées vers les simulateurs d'entraînement pour des

engins volants ou terrestres, puis se sont diversifiées dans des domaines tels que l'audiovisuel, le cinéma et l'art. Il est également important de noter que le traitement d'images repose sur des études liées à la structure des machines de traitement, car une image contient un grand nombre de données. Pour une image animée, il faut traiter $N \times M \times P$ échantillons par seconde (N points par lignes, M lignes et P images par seconde). Ainsi, le traitement d'images nécessite une puissance de calcul importante, ce qui exige des architectures très performantes avec un haut degré de parallélisme et une grande vitesse de traitement. [17]

I.5.4 Le codage d'images avec compression d'information :

Le codage d'images avec compression d'information est une technique de traitement d'images qui permet de réduire la quantité de données requises pour stocker ou transmettre une image numérique. Cela implique l'utilisation d'algorithmes de compression qui éliminent les redondances et les informations inutiles dans une image, tout en préservant la qualité visuelle de l'image. Les techniques de compression d'images peuvent être classées en deux catégories : la compression avec perte et la compression sans perte.

La compression avec perte permet une compression plus importante, mais peut entraîner une perte de qualité d'image, tandis que la compression sans perte permet de conserver une qualité d'image parfaite mais avec une compression moins importante. La compression d'images est largement utilisée dans les domaines de la communication numérique, la télévision, la photographie numérique, la médecine, l'archivage numérique, entre autres. [17]

I.6 Conclusion :

En conclusion, le traitement d'images et vidéos est un domaine en constante évolution qui a de nombreux domaines d'application, tels que la médecine, l'industrie cinématographique, la publicité et la sécurité. Nous avons vu que l'image numérique a des caractéristiques spécifiques et peut être de différents types, tels que les images bitmap et vectorielles. Nous avons également exploré les concepts clés de la vidéo, tels que les formats, les standards et les codages, ainsi que le montage vidéo. Enfin, nous avons examiné les différents types de traitements d'images et vidéos, tels que le traitement d'image, la reconnaissance de formes, la détection d'objets et la segmentation d'images. En somme, le traitement d'images et vidéos est un domaine fascinant et en constante évolution qui a de nombreuses applications potentielles dans notre vie quotidienne.

Chapitre

II

II.1 Introduction :

La compression de données est un domaine étendu qui a été abordé dans plusieurs ouvrages et articles. Actuellement, de nombreux travaux de recherche sont consacrés à ce sujet en raison des enjeux économiques qui y sont liés. La compression de données est principalement utilisée dans les applications informatiques. L'objectif de ce chapitre est de fournir un aperçu succinct de quelques principes fondamentaux de la compression de données. Et la compression d'image et vidéo

II.2 Définition de la compression :

La compression est le processus de réduction de la taille d'un fichier ou d'un ensemble de données sans altérer de manière significative les informations contenues dans le fichier. Elle est souvent utilisée pour économiser de l'espace de stockage ou pour accélérer la transmission de données sur un réseau. La compression peut être réalisée de manière lossless (sans perte) ou lossy (avec perte), selon le niveau de qualité acceptable pour les données compressées.[18]

II.3 Les objectifs de la compression :

- **Économiser de l'espace de stockage** : la compression permet de stocker plus de données sur un même espace de stockage. Cela est particulièrement utile pour les disques durs de taille limitée ou les supports de stockage portables comme les clés USB.
- **Réduire le temps de transfert des données** : la compression peut réduire la taille des données à transférer, ce qui réduit le temps nécessaire pour envoyer ou recevoir des fichiers sur un réseau. Cela est particulièrement important pour les transferts de gros fichiers ou pour les connexions internet lentes.
- **Améliorer les performances globales d'un système** : lorsque les données sont compressées, cela réduit la charge de travail du processeur, ce qui peut améliorer les performances globales du système. Cela est particulièrement important pour les systèmes à faible puissance, tels que les appareils mobiles ou les ordinateurs portables.
- **Permettre le stockage et le transfert de fichiers volumineux** : la compression permet de diviser des fichiers volumineux en plusieurs parties plus petites, ce qui les rend plus faciles à gérer et à transférer.

- **Augmenter la qualité de transmission des signaux audio ou vidéo** : la compression peut être utilisée pour réduire la taille des fichiers audio et vidéo sans perdre de qualité. Cela est particulièrement important pour la diffusion en direct ou en temps réel sur Internet.
- **Protéger les données contre la corruption ou la perte** : certains algorithmes de compression incluent des méthodes de détection et de correction des erreurs, ce qui permet de protéger les données contre les dommages ou la perte lors de leur stockage ou de leur transfert.

II.4 la compression des données:

Les différents algorithmes de compression reposent sur trois critères principaux :

- **Le taux de compression** : il s'agit du rapport entre la taille du fichier compressé et celle du fichier original. Ce critère permet d'évaluer la qualité de la compression.
- **La qualité de compression** : elle peut être avec ou sans perte, et dans le cas d'une compression avec perte, le pourcentage de perte est également pris en compte.
- **La vitesse de compression et de décompression** : ce critère est important pour mesurer l'efficacité du compresseur et du décompresseur.[19]

II.4.1 Types de Compression:

Compression est souvent considérée comme un algorithme permettant de réduire considérablement la taille des données (compression physique). Toutefois, il est également possible d'adopter une approche différente en considérant que l'objectif premier de l'algorithme de compression est de recoder les données dans une représentation différente et plus compacte contenant les mêmes informations (compression logique).

La distinction entre la compression physique et logique se fait sur la base de la façon dont les données sont compressées, ou plus précisément, comment les données sont réarrangées dans une forme plus compacte. Pour économiser de l'espace, il est parfois nécessaire de compresser les fichiers, mais le résultat n'est pas toujours optimal. Il est donc important de choisir en connaissance de cause entre deux types de compression :

la compression avec pertes

la compression sans pertes. [20]

II.4.1.1 Compression sans perte :

La compression sans perte est une technique de compression de données qui permet de réduire la taille d'un fichier sans perdre d'informations. Pour y parvenir, la méthode consiste à trouver des motifs répétitifs dans les données et à les remplacer par des symboles plus courts.[21]

II.4.1.1.1 Techniques de compression sans perte :

A - Le codage par plages (RLE) :

Le codage par plages (RLE), également appelé codage par répétition de symboles, est une technique de compression de données utilisée pour réduire la taille d'un fichier sans perte de données. Le principe de base du RLE est de remplacer une séquence de caractères identiques par un code qui représente le nombre de fois où ce caractère apparaît consécutivement. [19]

Par exemple, si l'on considère la séquence de caractères "AAAAABBBBCCCC", on peut remplacer cette séquence par "5A3B4C". Dans cet exemple, la séquence de cinq "A" est remplacée par "5A", la séquence de trois "B" est remplacée par "3B" et la séquence de quatre "C" est remplacée par "4C". Le résultat compressé est donc "5A3B4C".

Le RLE est particulièrement efficace pour compresser des fichiers qui contiennent des zones répétitives ou homogènes, tels que des images ou des vidéos. Cependant, il peut ne pas être efficace pour compresser des fichiers qui contiennent des données aléatoires ou des zones qui ne se répètent pas de manière significative.

En utilisant le RLE, on peut réduire la taille d'un fichier de manière significative, sans perdre aucune information. Cependant, le taux de compression dépend de la nature des données et le RLE est souvent utilisé en combinaison avec d'autres techniques de compression pour obtenir des taux de compression plus élevés.

B- Le codage de Shannon-Fano :

C'est une méthode de compression de données basée sur le principe du codage de Shannon, qui utilise des codes de longueur variable pour représenter des symboles différents. Cette méthode a été développée indépendamment par Claude Shannon et Robert Fano.[22]

Le principe de base du codage de Shannon-Fano est de diviser l'ensemble de données en sous-ensembles de manière récursive, en fonction des probabilités d'apparition de chaque symbole. Chaque sous-ensemble est ensuite codé à l'aide d'un code binaire de longueur

variable, en utilisant des codes courts pour les sous-ensembles avec les probabilités les plus élevées et des codes plus longs pour les sous-ensembles avec les probabilités les plus faibles.

Par exemple, si l'on considère un ensemble de données contenant les symboles "A", "B", "C" et "D", avec les probabilités d'apparition suivantes:

- A : 0,5
- B : 0,25
- C : 0,125
- D : 0,125

On peut diviser l'ensemble de données en deux sous-ensembles, en fonction des probabilités d'apparition :

- Sous-ensemble 1 : "A" et "B" (probabilité totale : 0,75)
- Sous-ensemble 2 : "C" et "D" (probabilité totale : 0,25)

Ensuite, on peut attribuer les codes binaires suivants à chaque sous-ensemble:

- Sous-ensemble 1 : 0
- Sous-ensemble 2 : 1

Ensuite, on peut diviser chaque sous-ensemble en sous-ensembles plus petits, en utilisant la même méthode, jusqu'à ce que chaque symbole soit représenté par un code binaire de longueur variable.

Le codage de Shannon-Fano est similaire au codage de Huffman, qui utilise également des codes de longueur variable pour représenter des symboles différents. Cependant, le codage de Shannon-Fano divise l'ensemble de données de manière récursive, tandis que le codage de Huffman utilise une méthode de construction de l'arbre de codage pour déterminer les codes de longueur variable.

C- Le codage de Huffman :

Le codage de Huffman est une méthode de compression de données qui utilise des codes de longueur variable pour représenter des symboles différents. Cette méthode a été développée par David Huffman en 1952

Le principe de base du codage de Huffman est de représenter les symboles les plus fréquents par des codes binaires plus courts, et les symboles moins fréquents par des codes binaires plus longs. Pour cela, on construit un arbre de codage binaire, où chaque feuille correspond à un symbole, et où les chemins de la racine aux feuilles représentent les codes binaires pour chaque symbole.

La construction de l'arbre de codage se fait en deux étapes : la création d'une file de priorité des symboles, en fonction de leur fréquence d'apparition dans l'ensemble de données, et la fusion des symboles les moins fréquents en paires, pour former des sous-arbres plus grands. [22]

Par exemple, si l'on considère un ensemble de données contenant les symboles "A", "B", "C", "D" et "E", avec les fréquences d'apparition suivantes :

- A : 5
- B : 3
- C : 1
- D : 1
- E : 1

On peut créer une file de priorité des symboles, en fonction de leur fréquence d'apparition :

- C : 1
- D : 1
- E : 1
- B : 3
- A : 5

Ensuite, on peut fusionner les symboles les moins fréquents en paires, pour former des sous-arbres plus grands :

- C et D sont fusionnés en un sous-arbre de fréquence 2
- E et le sous-arbre précédent sont fusionnés en un sous-arbre de fréquence 3
- B et le sous-arbre précédent sont fusionnés en un sous-arbre de fréquence 6

- A et le sous-arbre précédent sont fusionnés en un sous-arbre de fréquence 11

Enfin, on peut construire l'arbre de codage binaire à partir des sous-arbres fusionnés, en attribuant le code binaire "0" pour chaque branche à gauche, et le code binaire "1" pour chaque branche à droite. Les codes binaires pour chaque symbole sont alors déterminés en suivant le chemin de la racine à la feuille correspondante.

Le codage de Huffman est similaire au codage de Shannon-Fano, qui utilise également des codes de longueur variable pour représenter des symboles différents. Cependant, le codage de Huffman utilise une méthode de construction de l'arbre de codage pour déterminer les codes de longueur variable, tandis que le codage de Shannon-Fano divise l'ensemble de données en sous-ensembles de manière récursive.

II.4.1.2 Compression avec perte :

La compression avec perte est une méthode de compression de données qui permet de réduire la taille d'un fichier en éliminant certaines informations considérées comme moins importantes ou moins perceptibles par l'utilisateur final. Contrairement à la compression sans perte, la compression avec perte ne permet pas de retrouver le fichier d'origine à 100% après la décompression, car certaines informations ont été perdues pendant le processus de compression.[21]

La compression avec perte est souvent utilisée dans des situations où la taille du fichier est plus importante que la qualité de l'image ou du son, comme dans la diffusion de vidéos en streaming sur Internet ou la sauvegarde de fichiers multimédias sur un espace de stockage limité.

II.4.1.2.1 Techniques de compression avec perte :

a- Transformation en ondelettes continue (TOC)

La TOC (Transformation en Ondelettes Continue) est une méthode de décomposition d'une fonction F appartenant à l'espace $L^2(\mathbb{R})$ sur une base d'ondelettes choisie. Contrairement à la transformée en ondelettes discrète, la TOC utilise des paramètres de dilatation et de translation continus pour la décomposition de F . La fonction F est ainsi projetée sur une famille d'ondelettes choisies avec des paramètres continus, ce qui permet de représenter F dans un plan temps-fréquence continu. La TOC est donc utile pour analyser les signaux continus et représenter leur évolution fréquentielle au cours du temps. Elle est notamment utilisée en traitement de signal, en compression d'image et en reconnaissance de formes.

Elle est définie par l'équation :[23]

$$W(a, b) = TOC(f)(a, b) = \langle \Psi_{a,b}, f \rangle = |a|^{-1/2} \int_{-\infty}^{+\infty} f(x) \overline{\Psi\left(\frac{x-b}{a}\right)} dx \quad (\text{II.1})$$

Où $\overline{\Psi}$: Représente le conjugué complexe de Ψ

b- Transformation en ondelettes discrètes (TOD) :

La transformation en ondelettes discrètes (TOD) est une méthode qui permet de décomposer un signal numérique en ondelettes discrètes. Contrairement à la transformation en ondelettes continue (TOC), la TOD utilise une base finie et discrète d'ondelettes pour effectuer la décomposition. Cette méthode est largement utilisée dans le traitement numérique du signal pour l'analyse et la compression de données. Elle est également utilisée dans d'autres domaines tels que la reconnaissance de formes, la vision par ordinateur et l'analyse de séries temporelles.

La discrétisation de la DWT implique que les opérateurs de dilatation et de translation (a et b) prennent des valeurs discrètes suivant l'expression suivante : [23]

$$(a, b) = (a_0^j, kb_0 a_0^j) \quad (\text{II.2})$$

Où $(j, k) \in \mathbb{Z}^2$

Pour former ce qu'on appelle un réseau discret ; avec $a_0 > 1$ (le pas de dilatation) et $b_0 \neq 0$ (le pas de translation).

Ainsi, la nouvelle famille d'ondelettes s'établit par:

$$\Psi_{j,k}(t) = a_0^{-j/2} \cdot \Psi(a_0^{-j/2} t - kb_0) \quad (\text{II.3})$$

Et la DWT s'établit comme suit :

$$W(a, b) = \langle \Psi_{j,k}, f \rangle = a_0^{-jt} \int_{-\infty}^{+\infty} \Psi(a_0^{-jt} t - kb_0) f(t) dt \quad (\text{II.4})$$

c- Transformée en cosinus discrète « DCT »:

La Transformée en Cosinus Discrète (DCT) est une méthode de traitement numérique qui est appliquée à chaque bloc d'une image ou d'une vidéo. Elle est une variante de la Transformée de Fourier, qui décompose un bloc en une somme de fonctions cosinus oscillant à des fréquences différentes. Cette transformation permet de représenter chaque bloc par une carte de fréquences et d'amplitudes plutôt qu'en pixels et coefficients de couleur, où la valeur

d'une fréquence reflète l'importance et la rapidité d'un changement, tandis que la valeur d'une amplitude correspond à l'écart associé à chaque changement de couleur.

En appliquant la DCT à chaque bloc de $N \times N$ pixels, on obtient une matrice de $N \times N$ fréquences, qui permet de représenter chaque bloc de manière compacte et de réduire la taille du fichier d'image ou de vidéo tout en conservant une qualité acceptable. La DCT est largement utilisée dans les domaines de la compression d'image et de la vidéo, ainsi que dans le traitement du signal audio pour l'analyse spectrale et la compression audio.[24]

Formule pour calculer la DCT sur une matrice $N \times N$:

$$DCT(i, j) = \frac{1}{\sqrt{2}} C(i) C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} pixel(x, y) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right) \quad (\text{II.5})$$

$$C(x) = \frac{1}{\sqrt{2}} \text{si } x \text{ vaut } 0, \text{ et } 1 \text{ si } x > 0.$$

Formule pour calculer la IDCT sur une matrice $N \times N$:

$$pixel(x, y) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} c(i) c(j) DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right) \quad (\text{II.6})$$

$$C(x) = \frac{1}{\sqrt{2}} \text{si } x \text{ vaut } 0, \text{ et } 1 \text{ si } x > 0.$$

II.6 Compression d'image numérique :

La compression d'image est une technique qui permet de réduire la taille d'un fichier image en utilisant différents algorithmes de codage. L'objectif de la compression d'image est de réduire la quantité de données nécessaire pour stocker une image tout en préservant sa qualité visuelle. Les algorithmes de compression d'image peuvent être classés en deux catégories : la compression sans perte et la compression avec perte.[25]

II.6.1 Le principe général de la compression d'images

Le principe général de la compression d'images consiste à utiliser des méthodes et des règles pour réduire la quantité de données nécessaires pour stocker une image, sans perdre les informations essentielles. Cette technique est également appelée codage source ou réduction du débit binaire. En utilisant une paire de fonctions, l'une pour compresser les données et l'autre pour décompresser les données, la compression d'image permet de réduire la taille du fichier tout en préservant la qualité visuelle de l'image. La fonction de compression permet de réduire la quantité de données nécessaires pour stocker l'image, tandis que la fonction de décompression permet de reconstituer l'image originale à partir des données compressées.[25]

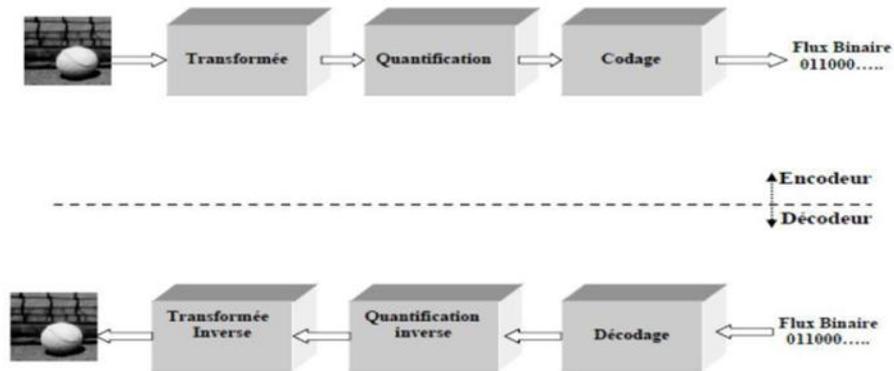


Figure II-1 :Schéma de compression /décompression classique d'image [25]

Un système de compression se décompose en trois modules que sont la transformée de l'image originale, la quantification des données de l'image et le codage des données quantifiées.

- **Première étape : La transformée** : Les pixels composant l'image numérique sont généralement corrélés ceci génère une information redondante qu'il faut exploiter pour la compression afin de diminuer la taille de l'image. En effet, en plus de réorganiser l'information, elle doit représenter les composantes importantes d'un signal avec le moins d'éléments possibles : c'est ce qu'on appelle donner une représentation creuse du signal ou, de manière équivalente, compacter l'énergie.
- **Deuxième étape : La quantification** : Dans le schéma de compression, l'étape de quantification est celle qui dégrade de manière irréversible l'image. Elle est cependant d'une importance capitale dans la réduction du débit binaire .Elle représente une étape clé de la compression. Elle approxime chaque valeur d'un signal par un multiple entier d'une quantité q , appelée quantum élémentaire ou pas de quantification.
- **Troisième étape Codage** : Le codage entropique est employé à la dernière étape de la chaine de compression avec pertes. Il consiste à donner la représentation binaire plus compacte de l'entité quantifiée, il existe différents types de codeur utilisés dans la compression d'image notamment, le codeur Hoffman et le codeur arithmétique.

II.6.2 Définition des critères utilisés en compression :

- **Taux de compression :**

Le taux de compression est un élément fondamental dans le domaine de la compression. C'est un outil de mesure qui évalue le degré de compression réalisé

Le taux de compression représente le rapport entre le nombre de bits utilisés par l'image originale et le nombre de bits utilisés par l'image compressée. [26]

Il est donc calculé comme suit:

$$\tau = \frac{\text{nombre de bit utilisés pour représenter l'image originale}}{\text{nombre de bit utilisés par l'image compressée}}$$

$$\tau = \frac{N \times M \times B}{N \times M} \quad (\text{II.7})$$

Où:

N, M : représentant respectivement le nombre de lignes et le nombre de colonnes de l'image.

B: désigne le nombre de bit utilisés pour la représentation d'un pixel

- Le débit :

Le débit binaire, qui est une autre mesure souvent utilisée, est déterminé à partir du nombre moyen de bits par pixel (bpp) de l'image codée, il est défini comme suit:

$$\text{débit} = \frac{\text{nombre de bit par pixel dans l'image originale}}{\text{taux}} \text{ Bit par pixel (Bpp)}$$

$$\text{débit} = \frac{\text{Bitstream}}{N \times M} \quad (\text{II.8})$$

- Mesure de distorsion :

La mesure de la distorsion est un sujet difficile. Il existe deux mesures couramment employées: l'erreur quadratique moyenne EQM (ou MSE pour Mean Square Error), qui est définie de la manière suivante:

$$\text{MSE} = \frac{\sum_{i=1}^N \sum_{j=1}^M (I(i,j) - \hat{t}(i,j))^2}{N \times M} \quad (\text{II.9})$$

I : L'image originale.

\hat{I} : L'image décodée.

M: le nombre de lignes de l'image.

N : le nombre de colonnes de l'image.

(i,j) : positionnement des pixels.

Et le rapport signal/bruit à son maximum : PSNR (Peak Signal Noise Ratio), qui est une métrique objective couramment utilisée pour mesurer la distorsion, son unité est le décibel (dB).

Il est défini par la relation suivante :

$$\text{PSNR} = 10 \log_{10} \left(\frac{(255)^2}{MSE} \right) \quad (\text{II.10})$$

Ou 255 est la valeur maximal d'un pixel pour une image codée par 8 bit/pixel en niveaux de gris.

Le rapport signal/bruit (SNR) est une autre variante de l'évaluation objective de la qualité, également exprimée en dB. Il a comme expression:

$$\text{PNS} = \frac{\sum_{i=1}^N \sum_{j=1}^M I^2(i,j)}{\sum_{i=1}^N \sum_{j=1}^M (I(i,j) - \hat{I}(i,j))^2} \quad (\text{II.11})$$

II.6.3 La décompression d'image :

La décompression d'image est le processus qui consiste à restaurer une image compressée à son état d'origine. Lorsque les images sont compressées, elles sont réduites en taille pour économiser de l'espace de stockage et améliorer la vitesse de transmission. Cependant, la compression peut altérer la qualité de l'image en éliminant des informations, ce qui peut causer des pertes de détails et une diminution de la résolution.

Pour restaurer l'image, il est nécessaire d'utiliser un algorithme de décompression. Il existe de nombreuses techniques de décompression d'image, telles que la décompression basée sur des prédictions, la décompression basée sur des transformées et la décompression basée sur des codages entropiques. Les algorithmes de décompression sont généralement basés sur les mêmes principes que ceux de compression, mais inversés.

En fin de compte, le choix de la méthode de décompression dépend du type de compression utilisé pour l'image, ainsi que des exigences en termes de qualité de l'image et de vitesse de traitement. [27]

II.6.4 Les norme de compressions d'image:

- **Le format JPEG:** est l'un des plus complexes et nécessite des connaissances mathématiques solides pour être compris dans son ensemble, mais malgré quelques désavantages, il offre des taux de compression très intéressants.

JPEG est une norme internationale (ISO 10918-1) pour la compression d'images fixes, en particulier d'images photographiques. La méthode de compression utilisée est "avec perte" et repose sur l'algorithme de transformée en cosinus discret DCT. Un mode "sans perte" a été développé plus tard, mais il n'a jamais vraiment été utilisé. Cette norme de compression a été développée par le comité JPEG (Joint Photographic Experts Group) et normalisée par l'ISO/JTC1 SC29. Ce type de compression est largement utilisé dans la photographie car il s'inspire des propriétés de perception visuelle de l'œil humain. [28]

Le format JPEG2000 est une norme internationale (ISO 15444-1). Il apporte certaines améliorations par rapport au format JPEG classique, permettant notamment des ajustements pour la compression sans perte ou la résistance aux erreurs de transmission. JPEG2000 implique une compression d'image basée sur le mécanisme de compression par ondelettes.

- **GIF :** GIF (Graphic Information Format) est un format léger qui peut également contenir des animations. Les images GIF ne peuvent contenir que 2, 4, 8, 16, 32, 64, 128 ou 256 des 16,8 millions de couleurs de la palette de couleurs en mode RVB. Il prend également en charge les couleurs transparentes.
- **PNG et MNG :** PNG (ISO 15948) pour Portable Web Graphics a été développé par le W3C pour remplacer le GIF. Il va au-delà de ce dernier, d'autant plus qu'il n'est pas limité à 256 couleurs. Encore une fois, le format est ouvert et permet une bonne compression sans perte. Il est recommandé de l'utiliser comme des GIF pour les petits logos. Côté photo, le poids de la photo n'est pas compétitif avec le format JPEG s'il permet une compression sans perte. Il convient de noter que PNG ne prend pas en charge l'animation, mais le format dérivé MNG est conçu pour cela. [28]

II.7 Compression de la vidéo numérique:

La compression de vidéo numérique est nécessaire pour réduire la quantité d'informations nécessaires à la transmission ou au stockage de la vidéo. Cela peut être réalisé en réduisant le débit binaire (bit rate) de la vidéo.

Les méthodes simples de réduction, telles que la réduction de la taille de l'image ou le nombre d'images par seconde, peuvent aider, mais ne suffisent souvent pas.

Les méthodes avancées de compression de vidéo peuvent être utilisées pour réduire considérablement la quantité de données nécessaires pour stocker ou transmettre une vidéo. Par exemple, une image contient souvent beaucoup d'informations redondantes, qui peuvent être éliminées pour réduire la taille du fichier. De plus, l'œil humain est moins sensible à la couleur qu'à la luminance de l'image. Il est donc possible de réduire l'information de chrominance sans que l'œil ne le remarque, ce qui permet également de réduire la quantité de données nécessaires pour stocker ou transmettre une vidéo.[11]

II.7.1 Codec:

Le Codec est un logiciel utilisé pour la compression et la décompression des fichiers vidéo. Le terme "Codec" est une contraction de "compresseur" et "décompresseur", ou encore "codeur" et "décodeur".

Le processus de compression consiste à utiliser un algorithme pour traiter la source vidéo et créer un fichier compressé, prêt pour la transmission ou le stockage. Pour lire ce fichier compressé, un algorithme inverse est appliqué pour obtenir une vidéo qui contient pratiquement le même contenu que la source vidéo d'origine. Le temps nécessaire pour compresser, envoyer, décompresser et afficher un fichier est appelé la latence. Plus l'algorithme de compression est perfectionné, plus la latence est élevée.

Les codecs vidéo de différentes normes ne sont généralement pas compatibles entre eux. En d'autres termes, le contenu vidéo compressé avec un algorithme ne peut pas être décompressé avec un autre algorithme. Par exemple, un décodeur MPEG-4 n'est pas compatible avec un encodeur H.264. Cela est dû au fait qu'un algorithme ne peut pas décoder correctement le résultat obtenu par un autre algorithme. Cependant, il est possible d'utiliser plusieurs algorithmes différents dans le même logiciel ou matériel, ce qui permet la coexistence de plusieurs formats.[11]

II.7.2 Technique de compression Vidéo :

II.7.2.1 Technique intra-image (compression spatiale) :

Les algorithmes de compression vidéo intra compression d'image où chaque image est travaillée individuellement. Les données sont réduites en supprimant les informations

redondantes ou inutiles car non remarque l'œil humain. Il existe plusieurs techniques. En voici quelques exemples

• **Transformation des couleurs :**

La technique de transformation des couleurs permet de modifier le système de stockage colorimétrique pour en utiliser un qui est moins exigeant en termes d'espace mémoire. Par exemple, le codec vidéo MPEG (1, 2 ou 4) peut convertir le système RVB classique (image de gauche), qui enregistre la valeur de chaque pixel pour le rouge, le vert et le bleu, en un système plus proche de la représentation de l'œil humain et qui nécessite moins d'espace de stockage, le système YUV, qui intègre une composante de luminosité et deux valeurs de chrominance.[11]

Les formules pour passer de l'espace RVB à l'espace YUV et inversement sont simples et correspondent à des produits matriciels, qui sont des changements de bases car il s'agit d'espaces vectoriels de dimension 3. Les formules sont les suivantes :[29]

- $Y = (0,257 * R) + (0,504 * G) + (0,098 * B) + 16$
- $Cr = V = (0,439 * R) - (0,368 * G) - (0,071 * B) + 128$
- $Cb = U = -(0,148 * R) - (0,291 * G) + (0,439 * B) + 128$

Et pour passer de l'espace YUV à l'espace RVB, les formules sont :

- $B = 1,164 * (Y - 16) + 2,018(U - 128)$
- $G = 1,164(Y - 16) - 0,813(V - 128) - 0,391(U - 128)$
- $R = 1,164(Y - 16) + 1,596(V - 128)$

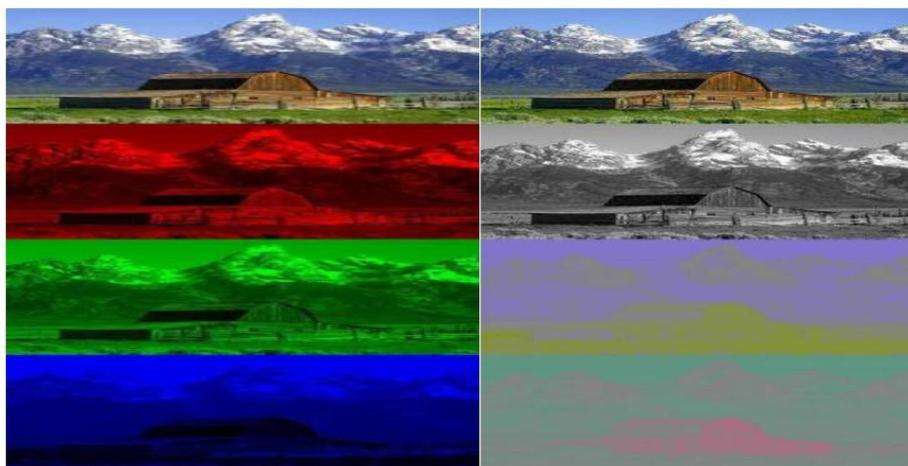


Figure II.2 : Exemple de technique intra-image par transformation des couleurs

- **Elimination les redondances :**

Une autre technique couramment utilisée dans la compression vidéo intra-image est la DCT (discrete cosine transform). Cette technique découpe chaque image en petits blocs et les transforme en une série de coefficients de fréquence. Les coefficients de haute fréquence sont souvent très petits et peuvent être négligés sans perte de qualité d'image significative. La DCT permet donc de réduire considérablement la quantité de données à stocker tout en maintenant une bonne qualité d'image. [11]

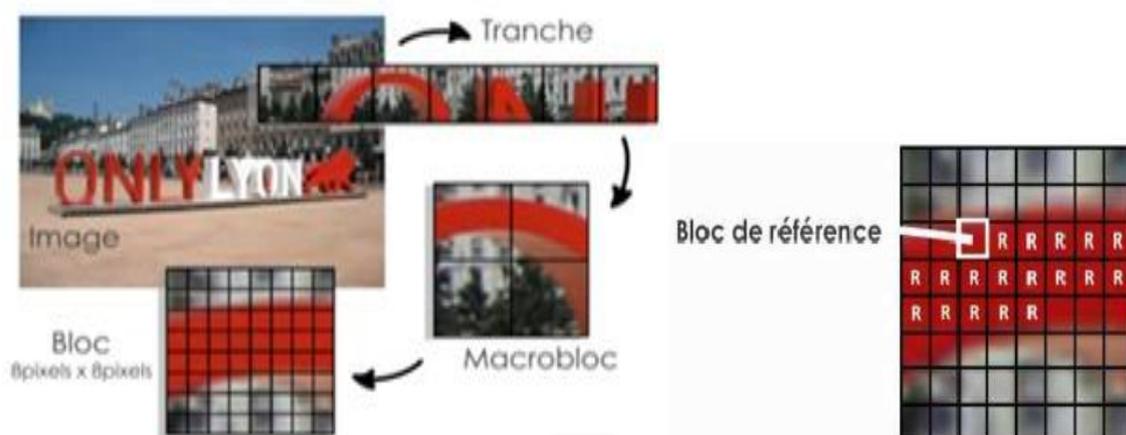


Figure II.3 : Exemple de technique intra-image par Elimination les redondances

En combinant ces différentes techniques, il est possible d'obtenir une compression vidéo intra-image efficace et de qualité. Cependant, pour atteindre des taux de compression encore plus élevés, les codecs vidéo modernes utilisent également des techniques de compression inter-image, qui exploitent les similitudes entre les images successives d'une séquence vidéo.

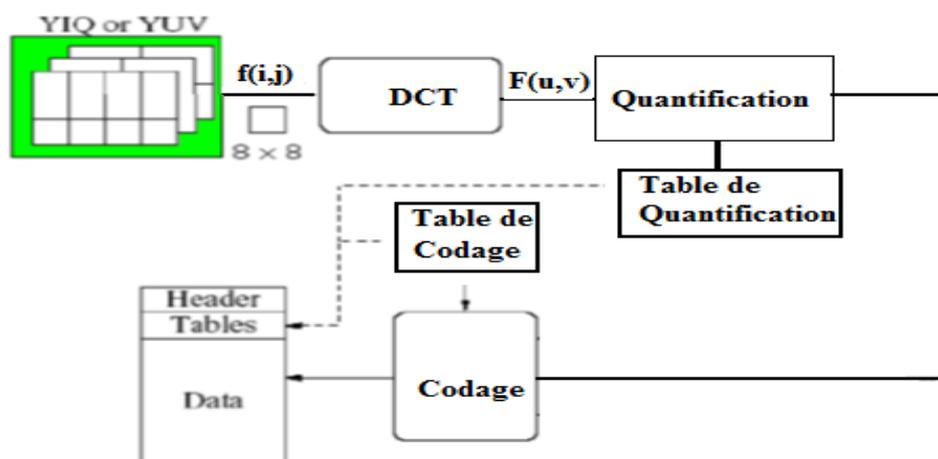


Figure II.4: Technique intra-image (compression spatiale)[30]

II.7.2.2 La compression temporelle :

La technique inter-image, également appelée compression temporelle, consiste à exploiter la corrélation temporelle entre les images successives d'une vidéo pour réduire la redondance de l'information. En effet, il est souvent possible de repérer des éléments fixes ou mobiles qui suivent un mouvement logique entre les différentes images. Ainsi, au lieu de stocker chaque image de manière indépendante, on ne stocke que la différence entre chaque image et la précédente. Les zones redondantes sont alors reprises des images précédentes, ce qui permet de réduire considérablement la taille du fichier vidéo. Différents mécanismes peuvent être utilisés pour identifier ces redondances et appliquer la compression temporelle.[11]

- **Le codage différentiel :**

Le codage différentiel est un mécanisme où une image est comparée à une image de référence et seuls les pixels qui ont changé par rapport à cette image de référence sont codés. Cela permet de réduire le nombre de valeurs codées et stockées pour chaque pixel. Le codage différentiel compare des éléments statiques d'une image à l'autre.

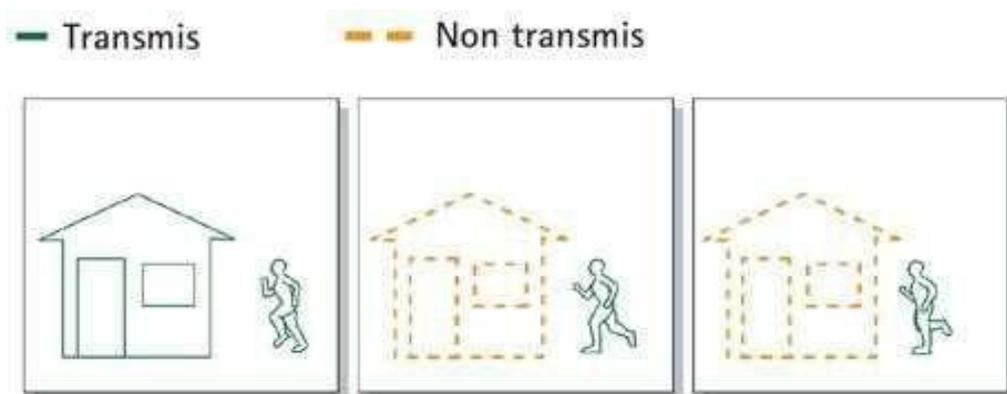


Figure II. 5 : Exemple sur la technique Inter-image par le codage différentiel

Comment comparer les blocs ? (Blocs Matching) :

- Trouver les blocs entre les frames $n+1$ et n ayant les intensités très proches
- Utiliser entre les blocs l'erreur moyenne quadratique (Mean Square Error):

$$MSE = \left(\frac{1}{N \times M} \right) \sum_{x,y} (n + 1(x + dx, y + dy) - I_n(x, y))^2 \text{ (II.12) [23]}$$

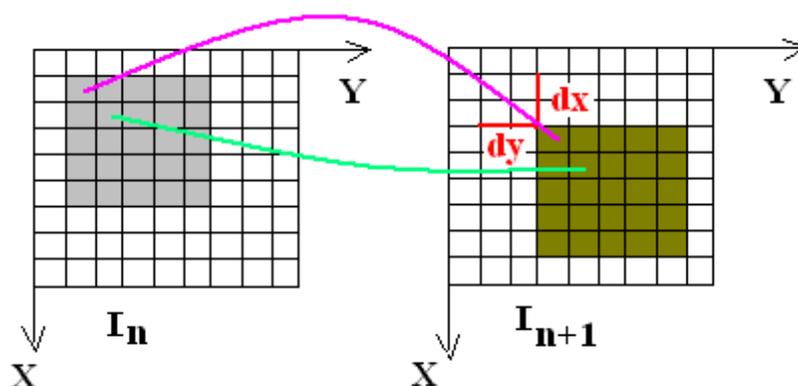


Figure II.6 : comparer les blocs

- **La compensation de mouvement :**

La compensation de mouvement est une technique utilisée dans le traitement de séquences vidéo. Elle repose sur le constat que beaucoup de ce qui compose une nouvelle image dans une séquence vidéo se retrouve dans une image adjacente, mais pas forcément au même endroit. Ainsi, il est possible de prédire une nouvelle image bloc par bloc en recherchant un bloc identique dans une image de référence. Si un bloc de référence est trouvé, l'encodeur code l'emplacement dans l'image de référence où se situe le bloc identique, ainsi que le vecteur de mouvement qui utilise moins d'espace que le codage du contenu réel d'un bloc. Cette technique permet donc de compresser efficacement les séquences vidéo en évitant la répétition d'informations identiques dans les images successives.[11]

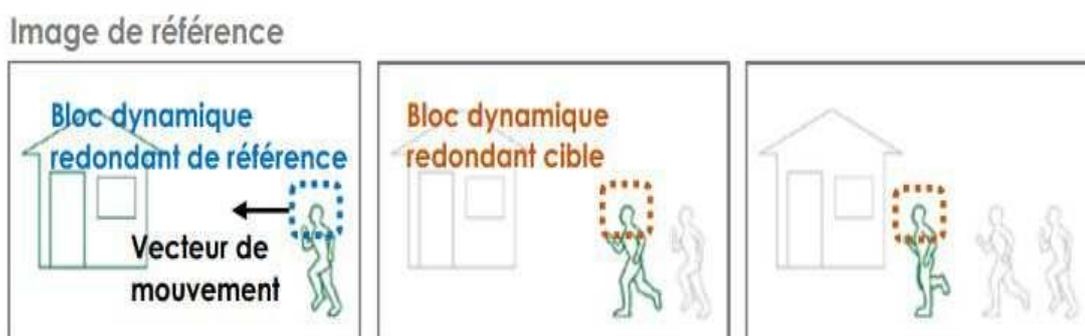


Figure II.7 : Exemple sur la technique Inter-image par La compensation de mouvement

La classification des images en différents types (I, P ou B) est réalisée grâce à la prédiction inter-images lors de la division des images vidéo en groupes de trames appelées GOP. Chaque GOP commence par une image de référence autonome (image I) suivie d'une série d'images prédictives (images P et B) qui sont codées en utilisant des informations des images I et/ou P précédentes et/ou suivantes. L'utilisation de cette technique permet une compression efficace des données pour chaque image, tout en préservant une qualité d'image acceptable.

Les images de référence (image I) sont les premières images de la séquence vidéo et sont généralement compressées spatialement de manière intra-image.

Les images prédictives (images P) nécessitent généralement moins de bits que les images de référence (image I), mais sont plus sensibles aux erreurs en raison de leur dépendance complexe vis-à-vis des images I et/ou P antérieures.

Les images inter-images bi-prédictives (images B) font référence à la fois à une image antérieure et à une image future et sont compressées en comparant les images I et/ou P précédentes ou suivantes pour trouver la plus ressemblante et définir les changements et les vecteurs de mouvement.[31]

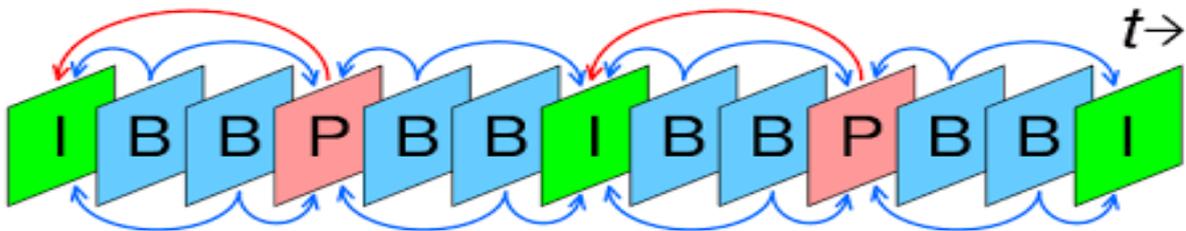


Figure II.8 La classification des images en différents types (I, P ou B)

La compression spatiale est souvent utilisée en combinaison avec la compression temporelle pour obtenir des taux de compression vidéo plus élevés. En utilisant ces deux techniques de compression, il est possible de réduire considérablement la taille des fichiers vidéo tout en préservant une qualité d'image acceptable

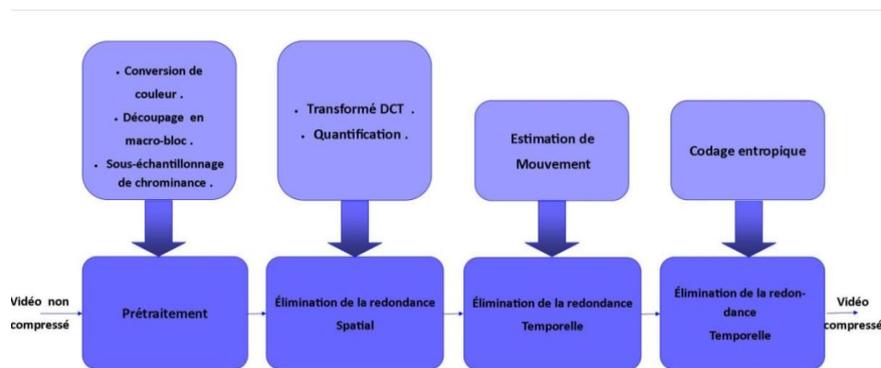


Figure II.9: Modules d'un système de compression vidéo [32]

II.7.3 Les normes de compression d'une vidéo :

Les normes vidéo sont caractérisées par leurs fréquences d'échantillonnage spatiale et temporelle, ainsi que leur structure d'entrelacement. Les applications liées à l'imagerie numérique nécessitent l'utilisation de signaux vidéo numériques différents en raison de contraintes telles que le débit, la capacité de stockage et l'affichage.

Pour les images fixes, la norme JPEG (Joint Photographic Experts Group) est utilisée, tandis que pour la compression vidéo, les normes MPEG développées par l'ISO et les normes H26* créées par l'ITU-T sont les plus courantes pour atteindre des facteurs de compression élevés.

Le groupe MPEG a été créé en 1988 par l'ISO pour la compression de séquences audiovisuelles. Le MPEG-1 a été créé pour les applications CD-I et CD-vidéo, tandis que le MPEG-2 a été lancé en 1995 pour être utilisé en TVHD (télévision haute définition). Enfin, le MPEG-4 a été publié en 1999 et est utile dans plusieurs applications.

Les normes H.261, H.263 et H.263+ ont été développées respectivement en 1990, 1996 et 1998. La norme H.261 est utilisée pour les applications de vidéo téléphonie sur lignes ISDN, tandis que le H.263 est utilisé pour les applications à très bas débit (inférieur à 64 kbits/s) comme la vidéoconférence. [33]

II.9 Conclusion :

En conclusion, la compression est une technique importante dans le domaine de l'informatique pour réduire la taille des données. Les différentes techniques de compression, y compris la compression sans perte et la compression avec perte, ont des avantages et des inconvénients en fonction de l'application. Dans le cas de la compression d'images et de vidéos, les normes sont essentielles pour garantir la compatibilité entre les différents équipements et logiciels. La compression est un domaine en constante évolution, avec de nouvelles techniques et normes émergentes régulièrement pour répondre aux besoins de stockage et de transmission de données toujours croissants.

Chapitre III

III.1 Introduction :

L'objectif de ce travail est de développer une approche générale pour la compression et la décompression de vidéos à l'aide de la transformée en cosinus discrète (DCT). Dans un premier niveau, nous cherchons à extraire les critères de performances tels que le taux de compression (CR), la distorsion, l'erreur quadratique moyenne, etc. Notre but est de trouver un bon compromis entre ces mesures, c'est-à-dire un fort rapport signal sur bruit (PSNR) et un taux de compression minimal, tout en préservant la qualité visuelle de la vidéo originale. De plus, nous devons respecter la contrainte de temps, car un système plus rapide est préférable.

Dans ce chapitre, nous appliquons la compression/décompression de vidéos en utilisant le programme MATLAB et obtenons diverses simulations. Nous explorons les différentes techniques de compression basées sur la DCT, telles que la quantification, le codage RLE et codage ZIGZAG , pour obtenir des résultats optimaux. Nous évaluons également les performances en termes de qualité visuelle et de taux de compression, en nous assurant que la vidéo comprimée conserve une qualité acceptable pour l'utilisateur final.

III.2 Définition de Matlab :

MATLAB est un langage de programmation de haute performance conçu spécifiquement pour les applications techniques. Il offre un environnement convivial qui combine le calcul, la visualisation et la programmation, permettant ainsi d'exprimer les problèmes et les solutions de manière mathématique, ce qui est familier à de nombreux utilisateurs. [34]

Les utilisations courantes de MATLAB incluent les domaines suivants : [34]

- Mathématiques et calculs : MATLAB fournit des fonctionnalités puissantes pour effectuer des opérations mathématiques complexes, telles que les calculs matriciels, les intégrations, les dérivations, etc.
- Développement d'algorithmes : MATLAB permet aux utilisateurs de développer et de tester des algorithmes, de les optimiser et de les implémenter dans des applications.
- Modélisation, simulation et prototypage : MATLAB offre des outils pour créer des modèles et des simulations de systèmes, tels que des modèles physiques, des systèmes de contrôle, des modèles financiers, etc. Il est également utilisé pour le prototypage rapide de nouvelles idées.

- Analyse, exploration et visualisation des données : MATLAB fournit des fonctionnalités avancées pour analyser et explorer des ensembles de données, effectuer des statistiques, créer des graphiques et des visualisations, afin de faciliter la compréhension des données.
- Graphiques scientifiques et techniques: MATLAB permet de créer des graphiques de haute qualité pour représenter visuellement des données, des fonctions, des modèles, etc. Il offre une large gamme d'outils de personnalisation pour créer des graphiques adaptés aux besoins spécifiques.
- Développement d'applications, y compris la création d'interfaces utilisateur graphiques : MATLAB offre des fonctionnalités pour développer des applications complètes, notamment la création d'interfaces utilisateur graphiques (GUI), permettant aux utilisateurs de construire des applications interactives avec des boutons, des graphiques, des champs de saisie, etc.

III.3 Transformation en cosinus discrète :

La transformée en cosinus discrète (DCT) est une transformation linéaire variante de la transformée de Fourier discrète, qui exprime une séquence finie de points de données en termes d'une somme de fonctions cosinus oscillant à différentes fréquences. Cette transformation, proposée pour la première fois par Nasir Ahmed en 1972, est largement utilisée dans le traitement du signal et la compression de données. Elle est appliquée à divers médias numériques tels que les images (comme JPEG et HEIF), les vidéos (comme MPEG et H.26x), l'audio (comme Dolby Digital, MP3 et AAC), Ainsi, la DCT permet d'obtenir une représentation fréquentielle purement réelle, ce qui facilite la compression des données tout en préservant une qualité acceptable. [35]

III.3.1 Variantes de la DCT :

La DCT type I (DCT-I) est utilisée pour les signaux symétriques et est principalement employée dans les applications de compression audio. Elle produit une séquence de coefficients DCT où le premier coefficient est la somme des échantillons du signal et les coefficients suivants représentent les différentes fréquences.

$$X_k = \frac{1}{2} (x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n \cos\left[\frac{\pi}{N-1} nk\right] \quad (\text{III.1})$$

for $k=0, \dots, N-1$.

La DCT type II (DCT-II), également connue sous le nom de DCT standard, est la plus couramment utilisée des quatre variantes. Elle est utilisée dans des applications telles que la compression JPEG et la compression vidéo MPEG. La DCT-II produit une séquence de coefficients où le premier coefficient représente la composante de basse fréquence et les coefficients suivants représentent les fréquences plus élevées.

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N} \left(n + \frac{1}{2}\right) k\right] \quad (\text{III.2})$$

for $k=0, \dots, N-1$

La DCT type III (DCT-III) est l'inverse de la DCT-II et permet de récupérer le signal d'origine à partir de sa représentation en coefficients DCT. Elle est utilisée dans les processus de décompression pour reconstituer le signal à partir des données compressées.

$$X_k = \frac{1}{2} x_0 + \sum_{n=1}^{N-1} x_n \cos\left[\frac{\pi}{N} \left(k + \frac{1}{2}\right) n\right] \quad (\text{III.3})$$

for $k=0, \dots, N-1$

Enfin, la DCT type IV (DCT-IV) est similaire à la DCT-II, mais avec une symétrie différente. Elle est principalement utilisée dans des applications telles que le codage audio perceptuel et la synthèse de signaux.

Ces quatre variantes de la DCT sont importantes et largement utilisées dans de nombreux domaines de traitement du signal et de compression de données. Chacune d'entre elles présente des propriétés uniques qui les rendent adaptées à des applications spécifiques. [35]

III.4 Architecture de Compression et la décompression DCT :

Le processus de compression et de décompression d'une vidéo donné dans la Littérature suit les étapes résumées par le schéma synoptique suivant :

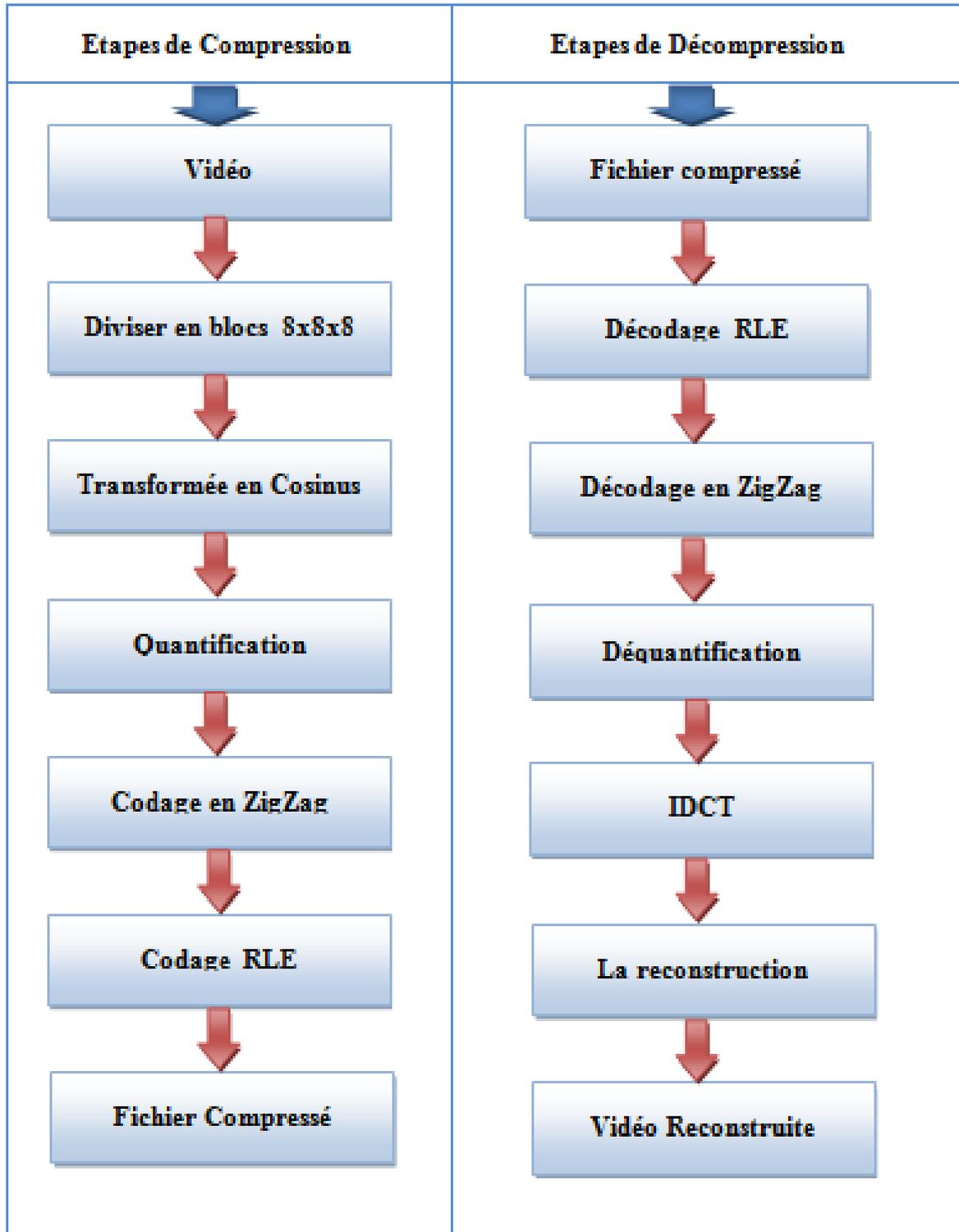


Figure III.1 :Architecture de Compression et la décompression

III.4.1 La compression et la décompression par la DCT :

Nous résumons les étapes de transformation, compression et décompression dans les sections suivantes:

III.4.1.1 Phase de Compression :

La phase de compression est divisée en plusieurs étapes :

Charger la vidéo :

Utilisez la fonction **VideoReader** pour charger la vidéo dans MATLAB.
Par exemple :

```
video = VideoReader('chemin_vers_la_video');[31]
```

Diviser en blocs 8x8x8 :

Lorsque nous parlons de diviser une vidéo en images de taille 8x8 avec une profondeur de 8 frames, nous utilisons une approche de découpage spatial et temporel. Le découpage spatial se réfère à la division de chaque image en blocs de taille 8x8 pixels dans les dimensions horizontale et verticale.

Chaque image est constituée de pixels qui représentent les différentes informations visuelles. En les divisant en blocs 8x8, nous obtenons de petits sous-ensembles de pixels qui peuvent être traités individuellement.

Le découpage temporel se réfère à la division des frames (images successives dans la vidéo) en blocs temporels de profondeur 8 frames. Cela signifie que nous considérons 8 frames consécutives comme une unité temporelle.

En combinant le découpage spatial et temporel, nous obtenons des blocs cubiques de taille 8x8x8. Chaque bloc représente une partie de l'image avec une profondeur de 8 frames. Ces blocs cubiques peuvent être traités indépendamment les uns des autres, permettant ainsi des opérations spécifiques sur chaque bloc.

Transformée en Cosinus Discrète (DCT) :

La transformation DCT est une technique couramment utilisée dans la compression vidéo pour convertir les informations spatiales d'un bloc en coefficients fréquentiels. Elle permet de représenter les variations de fréquence présentes dans un bloc d'image.

Pour chaque bloc 8x8x8 de la vidéo, nous pouvons appliquer la transformation DCT à chaque plan de profondeur. Cela signifie que nous appliquons la transformation DCT séparément aux blocs 8x8 dans chaque frame temporelle.

La transformation DCT convertit les valeurs des pixels d'un bloc en coefficients fréquentiels correspondant à différentes fréquences spatiales. Ces coefficients capturent les variations de fréquence présentes dans le bloc.

Exemple :

Imaginons que nous la divisons en blocs cubiques de taille 8x8x8. Voici le premier bloc :

Bloc 1 :

1	2	2	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
57	58	59	60	61	62	63	64

Le Tableau III.1 : Exemple de diviser une vidéo en images de taille 8x8.

Après avoir appliqué la DCT, nous obtenons les coefficients de fréquence suivants :

Coefficients de fréquence :

1368	0						
0	-104	0	0	0	0	0	0
0	0	-104	0	0	0	0	0
0	0	0	-104	0	0	0	0
0	0	0	0	-104	0	0	0
0	0	0	0	0	-104	0	0
0	0	0	0	0	0	-104	0
0	0	0	0	0	0	0	-104

Le Tableau III.2 : Exemple de Coefficients de fréquence

Dans cet exemple, les coefficients de fréquence représentent la contribution des différentes fréquences spatiales dans le bloc. Les valeurs non nulles indiquent la présence de certaines fréquences, tandis que les valeurs nulles indiquent l'absence de certaines fréquences.

Cela nous donne une représentation transformée du bloc, où nous avons extrait les caractéristiques fréquentielles. Ces coefficients de fréquence peuvent ensuite être utilisés pour la compression vidéo en quantifiant, encodant et stockant les valeurs de manière plus efficace

Quantification :

Après avoir obtenu les coefficients transformés par la DCT, la quantification permet de rejeter toute information qui paraît visuellement non significative. Puisque l'œil humain est sensible plutôt aux basses fréquences, il est indispensable de séparer les coefficients à basses fréquences de ceux qui représentent les hautes fréquences. C'est ainsi que la quantification est appliquée après la DCT de sorte à pouvoir adapter la quantification en conséquence.

Pour la quantification, nous utilisons une table de quantification prédéfinie. Cette table est déterminée en fonction du facteur de qualité souhaité pour la compression vidéo. Elle permet de réduire la précision des coefficients de fréquence afin de les rendre plus compressibles.

Supposons que nous utilisons une table de quantification simple pour l'exemple, donnant les valeurs suivantes :

Table de quantification :

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	123	1001
72	92	95	98	77	100	103	99

Le Tableau III.3 : Exemple de Table de quantification

Maintenant, nous appliquons la quantification en divisant chaque coefficient de fréquence par la valeur correspondante de la table de quantification, puis en arrondissant le résultat à l'entier le plus proche :

Coefficients quantifiés :

86	0	0	0	0	0	0	0
0	-9	0	0	0	0	0	0
0	0	-6	0	0	0	0	0
0	0	0	-4	0	0	0	0
0	0	0	0	-2	0	0	0
0	0	0	0	0	-1	0	0
0	0	0	0	0	0	-2	0
0	0	0	0	0	0	0	-1

Le Tableau III.4 : Exemple de Coefficients quantifiés

Les coefficients quantifiés représentent maintenant une version réduite et approximative des coefficients de fréquence d'origine. Ces valeurs plus petites permettent de réduire la quantité d'informations à stocker et à transmettre, contribuant ainsi à la compression de la vidéo.

Codage en ZigZag:

L'étape suivante après la quantification est l'application du parcours en zigzag sur les coefficients quantifiés. Le parcours en zigzag réorganise les coefficients de manière à regrouper les valeurs importantes et les valeurs négligeables, ce qui facilite leur compression et leur stockage.

Reprenons les coefficients quantifiés obtenus précédemment Appliquons le parcours en zigzag sur ces coefficients :

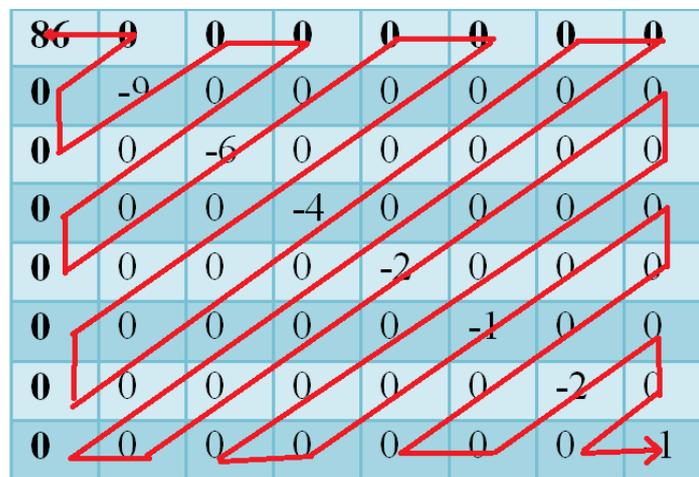


Figure III.2: Exemple de zigzag sur la table de quantification

Parcours en zigzag :

86	0	0	0	-9	0	0	0	0	0	0	0	-6	0	0	0	0	0	0	0
----	---	---	---	----	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---

0	0	0	-4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	0
---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----	---

0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	-2	0	0	0	-1
---	---	---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	----	---	---	---	----

Le parcours en zigzag réarrange les coefficients quantifiés en une séquence linéaire. Les premiers coefficients de la séquence sont généralement les plus importants, tandis que les derniers coefficients sont souvent négligeables. Cela permet d'exploiter la redondance des coefficients et de faciliter leur compression.

La séquence obtenue après le parcours en zigzag est utilisée pour l'étape suivante de la compression vidéo, qui est l'encodage des données.

Codage RLE

Après avoir effectué le parcours en zigzag, nous pouvons appliquer le codage RLE (Run-LengthEncoding) sur la séquence obtenue. Le codage RLE permet de compresser les données en remplaçant les répétitions successives d'une même valeur par une paire (valeur, nombre de répétitions).

Exemple :

Poursuivons avec l'application de l'encodage RLE (Run-LengthEncoding) sur la séquence obtenue après le parcours en zigzag.

Voici la séquence après le parcours en zigzag :

86	0	0	0	-9	0	0	0	0	0	0	0	-6	0	0	0	0	0	0	0
----	---	---	---	----	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---

0	0	0	-4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	0
---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----	---

0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	-2	0	0	0	-1
---	---	---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	----	---	---	---	----

Appliquons l'encodage RLE sur la séquence :

Encodage RLE :

[(86, 1), (0, 3), (-9, 1), (0, 7), (-6, 1), (0, 11), (-4, 1), (0, 14), (-2, 1), (0, 11), (-1, 1), (0, 7), (-2, 1), (0, 3), (-1, 1)]

Dans cette séquence encodée, chaque paire représente une valeur suivie du nombre de fois où cette valeur se répète consécutivement. Par exemple, (0, 6) signifie que la valeur 0 se répète 6 fois de manière consécutive.

Le codage RLE permet de réduire la taille des données en exploitant la redondance des répétitions dans la séquence. Cette séquence encodée est ensuite utilisée pour l'étape finale de la compression vidéo, qui est l'encodage des données compressées dans un format spécifique, tel que le format MPEG.

III.4.1.2 Phase de décompression :

La phase de décompression est l'inverse de la phase de compression. Elle est divisée comme suit :

Extraction des paramètres de compression :

Elle s'effectue par l'ouverture du fichier compressé afin d'extraire les paramètres (reconstruction de code binaire, les valeurs Application et Implémentation (DCTmax, DCTmin, Q, bloc, taille du fichier original,...))

Décompression de l'encodage RLE : Nous parcourons la séquence encodée et répétons chaque valeur selon la longueur spécifiée. Cela restaure la séquence après l'encodage RLE.

Déplacement en zigzag : Nous prenons la séquence décompressée et réorganisons les coefficients dans leur ordre d'origine en utilisant le parcours en zigzag inverse.

Déquantification : Nous multiplions chaque coefficient par la valeur correspondante de la table de quantification utilisée précédemment. Cela restaure les coefficients quantifiés à leurs valeurs approximatives d'origine.

Inverse de la DCT : Nous appliquons l'inverse de la DCT sur chaque bloc de coefficients pour obtenir les valeurs d'intensité d'origine. Ces coefficients servent à calculer l'erreur quadratique moyenne MSE et le rapport signal sur bruit crête PSNR

Reconstruction : Nous assemblons les blocs de pixels pour reconstruire Les images de la vidéo d'origine.

Remarque:

Il est important de noter que la conversion des couleurs peut être effectuée avant ou après la DCT, en fonction du besoin spécifique de l'application de compression vidéo. Cependant, la DCT elle-même ne nécessite pas de conversion de couleurs et peut être utilisée pour compresser les données vidéo directement à partir de l'espace de couleurs d'origine.

De plus, la conversion des couleurs facilite l'exploitation de la corrélation spatiale entre les pixels. En convertissant les couleurs avant la DCT

III.5 Tests et Résultats :

III.5.1 Logiciel de l'implémentation :

L'algorithme mentionné a été mis en œuvre sous Matlab 2010a et Matlab 2016a avec un PC Intel(R) Core (TM) i7- CPU ; 2.9GHz PC : 8Go de RAM utilisant. et PC Intel(R) Core (TM) i7- CPU ; 3.7 GHz PC : 16 Go de RAM utilisant.

III.5.2 Paramètres d'évaluation :

Les paramètres les plus utilisés dans le domaine de la compression vidéo sont similaires à ceux de la compression d'images, mais ils sont adaptés pour prendre en compte les caractéristiques spécifiques des vidéos. Parmi ces paramètres, on retrouve le rapport signal sur bruit crête (PSNR), le taux de compression (exprimé en Bpp : bits par pixel) ou en Mbps (mégabits par seconde), le gain de compression, ainsi que le temps de compression et de décompression. Ces mesures permettent d'évaluer la qualité visuelle de la vidéo, la quantité de données nécessaires pour la représenter et les performances temporelles de compression et de décompression.

III.5.3 Test 1:

La première teste se fait sur une vidéo trafic en niveaux de gris :

	frames sélectionnées de vidéo originale	frames sélectionnées de vidéo reconstruite
Fra me 1		
Fra me 5		
Fra me 8		

Tableau III.5 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.

Temps de compression	32.225103 seconds
Temps de décompression	228.256075 seconds
Taux de compression	12.68 %
Gain de compression	87.32%
MSE	1.9872
PSNR	48.5203dB

Tableau III.6: représente Paramètres d'évaluation de test 1

III.5.4 Test 2 :

La deuxième teste se fait sur une vidéo à la surface de la lune :

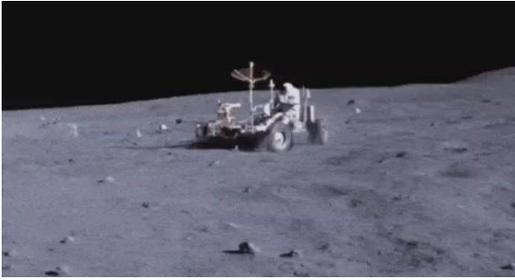
	frames sélectionnées de vidéo originale	frames sélectionnées de vidéo reconstruite
Frame 1		
Frame 5		
Frame 8		

Tableau III.7 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.

Temps de compression	13.690324 seconds
Temps de décompression	59.876872 seconds
Taux de compression	14.3 %
Gain de compression	85.87 %
MSE	3.8176
PSNR	42.0469

Tableau III.8 : représente Paramètres d'évaluation de test 2

III.5.5 Test 3 :

La troisième teste se fait sur une trafic accident :

	frames sélectionnées de vidéo originale	frames sélectionnées de vidéo reconstruite (1)	frames sélectionnées de vidéo reconstruite (2)
Frame :23			
Frame :30			
Frame 133			

Tableau III.9 : Échantillons de trames sélectionnées des vidéos utilisées avant et après compression.

Temps de compression	322.245103 seconds	280.275151 seconds
Temps de décompression	576.256075 seconds	483.365412seconds
Taux de compression	21.3 %	10.91 %
Gain de compression	78.7 %	89.09 %
MSE	0.8827	5.5467
PSNR	66.6233 dB	39.784 dB

Tableau III.10 :Représente Paramètres d'évaluation de test 3

III.5.6 Interprétations des tests :

Dans la première test , nous avons utilisé une vidéo en niveaux de gris avec une résolution moyenne

La compression était généralement bonne pour la taille, mais l'apparition d'une certaine distorsion était due au manque de qualité vidéo d'origine

Dans la deuxième test, nous avons utilisé une vidéo couleur avec une résolution moyenne. La compression était bonne, avec de très petites différences entre la vidéo d'origine et la vidéo reconstruite. La raison est due à la cohérence des couleurs. Il n'y avait pas beaucoup de couleurs dans la vidéo pour l'œil humain, et aussi grâce à la conversion des couleurs avant compression. et après la décompression

Dans la troisième test, nous avons combiné deux tests sur la vidéo haute définition en modifiant le facteur de qualité requis pour la compression vidéo.

- Le premier partie , la compression était très bonne en termes de clarté de la vidéo et généralement acceptable en termes de taille de vidéo reconstruite
- La deuxième partie, nous avons augmenté davantage la quantification pour obtenir une plus grande compression en termes de taille de fichier

Regardez les résultats joints aux tableaux pour chaque test

En général, la qualité de la compression et le temps de compression et de décompression sont liés à la taille et à la qualité du fichier d'origine Nous notons également que les outils utilisés dans les tests affectent les résultats de la compression et de la décompression.

III.6 Conclusion :

En conclusion, nous avons exploré le processus de compression et de décompression vidéo utilisant la transformation en cosinus discrète (DCT). Nous avons commencé par définir MATLAB comme un outil essentiel dans notre application. Ensuite, nous avons examiné les variantes de la DCT et leur utilisation dans la compression vidéo.

Dans la phase de compression, nous avons chargé la vidéo, divisé en blocs 8x8x8, appliqué la DCT pour convertir le domaine spatial en domaine fréquentiel, quantifié les coefficients, utilisé le codage en zigzag et le codage RLE pour compresser les données.

La phase de décompression a impliqué l'extraction des paramètres de compression, la décompression de l'encodage RLE, le déplacement en zigzag inverse, la déquantification et l'application de l'inverse de la DCT pour reconstruire la vidéo.

Nous avons également évalué les performances de la compression vidéo par la DCT à travers des tests. Trois tests ont été présentés pour illustrer les résultats obtenus.

En général, la compression vidéo par la DCT offre une méthode efficace pour réduire la taille des fichiers vidéo tout en maintenant une qualité visuelle acceptable. Cette technique est largement utilisée dans diverses applications, telles que la diffusion en continu, la télévision numérique et le stockage de vidéos sur des dispositifs avec des ressources limitées. La compression par la DCT est un domaine de recherche actif, et des améliorations continues sont apportées pour optimiser la compression et la qualité des vidéos compressées.

*Conclusion
Générale*

Conclusion générale :

Dans ce mémoire, nous avons exploré la méthode de compression de vidéos par la transformation en cosinus discrète (DCT). Nous avons commencé par introduire les concepts de base du traitement d'images et de vidéos, en mettant l'accent sur la compression des données comme un élément clé pour réduire la taille des fichiers et faciliter leur manipulation. Nous avons ensuite examiné les différentes techniques de compression, en soulignant l'importance de la compression par la DCT dans le domaine de la compression vidéo.

Au cours de notre étude, nous avons présenté les principes fondamentaux de la compression d'images et de vidéos, en mettant en évidence les objectifs de la compression et les critères utilisés pour évaluer sa qualité. Nous avons discuté des différents types de compression, en nous concentrant sur la compression sans perte et la compression avec perte, et en explorant les techniques spécifiques utilisées dans chaque cas. Nous avons également examiné les normes et les codecs utilisés dans la compression vidéo, ainsi que les techniques de compression spatiale et temporelle.

En ce qui concerne la compression par la DCT, nous avons défini la DCT et ses variantes, et nous avons présenté une architecture de compression et de décompression basée sur cette méthode. Nous avons décrit les phases de compression et de décompression, en expliquant comment la DCT est appliquée pour réduire la redondance et la quantité d'informations non essentielles dans les vidéos. Nous avons réalisé des tests et des évaluations à l'aide d'un logiciel d'implémentation, en utilisant des paramètres spécifiques pour mesurer l'efficacité et la qualité de la compression par la DCT.

Les résultats obtenus à partir de nos tests ont démontré que la compression par la DCT permet une réduction significative de la taille des vidéos tout en préservant une qualité visuelle acceptable. Cependant, il est important de noter que le choix des paramètres de compression et les compromis entre la taille et la qualité sont des éléments essentiels dans l'application pratique de cette méthode.

En générale, la compression par la DCT est une méthode efficace pour réduire la taille des vidéos tout en maintenant une qualité visuelle acceptable. Cette méthode trouve des applications dans de nombreux domaines tels que la diffusion en continu, la télévision numérique, les systèmes de surveillance, etc. Cependant, il reste encore des possibilités d'amélioration et de recherche dans ce domaine, notamment en explorant de nouvelles variantes de la DCT, en optimisant les paramètres de compression et en évaluant l'impact de la compression sur différents types de vidéos.

Conclusion Générale

Ce mémoire a permis de jeter les bases d'une compréhension approfondie de la compression vidéo par la DCT. Il est souhaitable que les travaux futurs se concentrent sur des domaines connexes tels que l'optimisation des performances, l'adaptation aux différents types de vidéos et l'exploration de nouvelles techniques de compression. Ces efforts contribueront à l'évolution de la compression vidéo et à l'amélioration des méthodes existantes pour répondre aux besoins croissants en matière de stockage, de transmission et de traitement des vidéos.

BIBLIOGRAPHIE

BIBLIOGRAPHIE

- [1] B. Bouamer, Y. Dehane , "Compression D'image Par La Méthode Tcd (Transformée En Cosinus Discrète) ", Université De Ghardaïa 2019/2020.
- [2] <https://Www.Netexplorer.Fr/Blog/Definition-Compression-De-Donnees>
- [3] Ahmed, N., Natarajan, T., & Rao, K. R. (1974). Tdc : Une Nouvelle Méthode De Transformation Pour Le Codage Des Images. Proceedings Of The Ieee, 65(1), 206-211.
- [4] M. Andre, "Introduction Aux Techniques De Traitement D'images", Eyrolles 1987
- [5] C. Taouche , " Stéganographie D'une Images Numérique " , Université Mentouri Constantine 2005.
- [6] https://Helpx.Adobe.Com/Ca_Fr/Photoshop/Using/Image-Size-Resolution.Html
- [7] <https://Www.Viewsonic.Com/Library/Fr/Tech-Fr/Resolution-Decran-Et-Format-Dimage/>
- [8] <https://Blog.Photo24.Fr/Bruit-Numerique-Quest-Ce-Que-Cest/>
- [9] H. M. Boucetta, " Détection Poursuite Et Comptage D'objet Par La Vision Artificielle" , Université De Blida, 2017.
- [10] A. A. Nair, " Hybridation Des Réseaux De Neurones Avec Les Essaims De Particules (Mlp-Pso) :Application A La Vérification De La Signature " , Université Des Sciences Et De La Technologie D'oran 2011.
- [11] F. Kada, A.Zahaf," Implémentation Des Algorithmes De Traitement Des Images Et Vidéos En Utilisant La Bibliothèque Opencv " , Université Abou Bakrbelkaid Tlemcen 2016.
- [12] <https://Www.Claireetclaire.Fr/Que-Sont-Les-Images-Bitmap-Les-Images-Vectorielles/>
- [13] <https://Routernation.Com/Guide-Du-Debit-Binaire-Video-Pour-La-Diffusion-En-Direct/>
- [14] https://En.Wikipedia.Org/Wiki/Video#Video_Formats
- [15] <http://Primatice.Phpnet.Org/Logiciels/Chromoweb/Aide/Codage.Htm>
- [16] <https://Www.Techno-Science.Net/Glossaire-Definition/Montage-Video.Html>
- [17] https://Unit.Eu/Cours/Videocommunication/Intro_Systemes_Comm.Pdf
- [18] <https://Adnethique.Org/Glossaire/Compression-De-Donnees>
- [19] http://Igm.Univ-Mlv.Fr/~Dr/Xpose2002/Compression/Default_Fichiers/Page0003.Htm
- [20] A.M. Bellabidi, "Compression D'image Sans Pertes Par Jpeg-Ls" , Université Mohamed Khider Biskra 2013.
- [21] <https://Www.Lemagit.Fr/Definition/Compression-Des-Donnees>
- [22] M. Elbar, M. Laouar, "Compression D'Image Par Fusion Technique ", Université De Ghardaïa 2019/2020.
- [23] M. Zeghib, A. Serouti Compression Des Images Avec Curvelet , Université Echahid Hama Lakhdar – El Oued 2014-2015
- [24] <https://Membres-Ljk.Imag.Fr/Valerie.Perrier/Siteweb/Node9.Html>
- [25] P. Steven. "Contribution A La Compression De Données". Thèse De Doctorat : Informatique. Montréal, 2001

BIBLIOGRAPHIE

- [26] Y. Boudjit " Compression D'images Sans Perte Par Des Techniques Du Codage Source",
Université Mohammed Seddik Benyahia – Jijel
- [27] Rafael C. Gonzalez, Richard E. Woods Et Steven L. Eddins, "Digital Image Processing
Using Matlab", Pearson Prentice-Hall, 2004.
- [28] R. Bahloul, " Factorisation Matricielle Non Negative Pour Larecognition Faciale "
,Université Abdelhamid Ibn Badis –Mostaganem, 2018-2019.
- [29] Travail D'étude, "Techniques De Compression Video Des Standards Mpeg "Universite De Nice-
Sophia Antipolis, 2004
- [30] S. Larabi, Courssystemesmultimedia, Master Rsd, Usthb,2014/2015
- [31]<https://Efreidoc.Fr/M1/Principes%20de%20%27audiovisuel/Cours/2012-13/Cours%20complet/2012-13.Cours.07.Compression-Des-Videos.Audiov.Pdf>
- [32] A.Bouzenad"Étude Des Différentes Normes De Compression Vidéo",Ecole Nationale
Polytechnique,2013.
- [33] M. Barlaud , C. Labit, "Compression Et Codage Des Images Et Des Vidéos". Paris : Hermes
Science, 2002
- [34] <https://Fr.Mathworks.Com/>
- [35] https://En.Wikipedia.Org/Wiki/Discrete_Cosine_Transform