

وزارة التعليم العالي والبحث العلمي

BADJI MOKHTAR- ANNABA UNIVERSITY
UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار - عنابة

Année : 2019

Faculté: Sciences de l'Ingénierat
Département: Electronique

MEMOIRE

Présenté en vue de l'obtention du diplôme de : MASTER

Intitulé :
**Evaluation de la transmission de la vidéo compressée par
la plateforme EvalVid**

Domaine : Sciences et Technologie
Filière : Télécommunications
Spécialité : Réseaux et télécommunications

Par :
Nouar Mohamed Seifeddine

DEVANT Le JURY

Président : Mr. NASRI	MAB	UBMA
Directeur de mémoire : Mr. KOUADRIA	MCA	UBMA
Co-directeur de mémoire: Mme HARIZE	MCA	UBMA
Examineurs : Mr. BOUTALBI	MAA	UBMA
Mr. AFIFI	PROF	UBMA

Dédicace :

Je dédié ce modeste travail à

La lumière de ma vie, au cœur le plus tendre et le plus doux, a celle qui
a tellement sacrifié pour me voir toujours meilleur : ma très chère

mère

A l'être le plus cher à mon cœur, a celui qui m'a toujours guidée par ses
conseils et qui m'a encouragée à poursuivre mes études : Mon père

A Mes chers frères : Ilyes, Aimen

Mes amies les plus proches : Mohamed, Islam, Ahmed, Khaled, Nour

Tous mes camarades de classes que j'aime beaucoup

A Mes meilleurs professeurs que je vénère et que je respecte

Remerciements :

Je remercie tout d'abord Dieu le tout puissant, le miséricordieux qui m'a donnée la force, la volonté et le courage afin d'élaborer ce modeste travail.

Je ne sais comment remercier mes parents qui m'a aidé moralement, financièrement, et qui étaient toujours patients, compréhensifs envers moi et qui ont su par leur tendresse et leur sacrifices me mettre sur les bons rails.

Je tiens à remercier Dr. KOUADRIA Nasreddine, pour avoir accepté de diriger ce travail malgré ses multiples occupations et le doctorant Islam Maansri et Sakhri aya qui m'ont aidé dans la partie théorique et pratique de ce travail.

Mes remerciements vont également à mes directeurs de stage, Mr Tounssi et Mr Chaouch, qui se sont rendu disponible pour me guider durant tout mon stage.

Mon grand respect à mes examinateurs et aux membres du jury

Enfin, je remercie tous ceux qui de près ou de loin ont apporté leur contribution pour l'achèvement de ce modeste travail.

Résumé :

Les technologies de l'information connaissent un développement significatif en termes de réduction des volumes des fichiers, c'est ce qu'on appelle compression. Ce domaine comporte plusieurs solutions pour réduire la taille de données toutes sortes: son, image et vidéo.

La compression multimédia réduit la taille du fichier et fait ainsi des économies d'espace disque et accélère la transmission. Il y a plusieurs encodeurs vidéo standards, citons pour l'exemple MPEGx, JM et HM. Ce mémoire de Master, traite une compression/transmission vidéo par JM. L'étude de l'impact de la transmission sur la vidéo compressée est analysée à l'aide l'outil logiciel EvalVid.

Mot clés : Multimédia, compression vidéo, EvalVid, JM

Abstract :

Information technologies are experiencing a significant development in terms of reducing file volumes, this is called compression. This domain has several solutions to reduce the size of data of all kinds: sound, image and video.

Multimedia compression reduces file size, saving disk space and speeding up transmission. There are several standard video encoders, for example MPEGx, JM and HM. This master thesis deals with video compression / transmission by JM. The study of the impact of the transmission on compressed video is done by the EvalVid software tool.

Keywords: Multimedia, video compression, EvalVid, JM

ملخص :

تشهد تقنيات المعلومات تطوراً ملحوظاً من حيث تقليل الأحجام ، وهذا ما يسمى بالضغط. يحتوي هذا المجال على العديد من الحلول لتقليل حجم البيانات بجميع أنواعها: الصوت والصورة والفيديو.

يعمل ضغط الوسائط المتعددة على تقليل حجم الملف وتوفير مساحة القرص وزيادة سرعة النقل. هناك العديد من برامج ضغط الفيديو المقننة ، على سبيل المثال MPEGx و JM و HM. تتناول هذه الرسالة ضغط الفيديو JM. تتم دراسة تأثير الإرسال على الفيديو المضغوط بواسطة أداة البرنامج EvalVid.

كلمات مفتاحية : Multimedia, video compression, EvalVid, JM .

Liste des abréviations :

DPCM : Differential pulse-code modulation.

ADPCM : Adaptive Differential Pulse-Code Modulation.

GIF : Graphic Interchange Format.

PNG : Portable Network Graphics.

MPEG : Moving Picture Experts Group.

AVI : Audio Video Interleave.

WMV : Windows Media Video.

ITU : International Telecommunication Union.

ISO : International Organization for Standardization.

CEI : International Electrotechnical Commission.

DVD : Digital VideoDisc.

HEVC : High Efficiency Video Coding.

MSE : Mean Square Error.

PSNR: peak signal-to-noise ratio.

SSIM : Structural Similarity Index.

AVC : Advanced Video Coding.

AI : All Intra.

RA : Random Access.

LD : Low Dealy.

QP : taux de compression.

SDP : Session Description Protocol.

PLR : Packet loss ratio.

Listes des tableaux :

Tableau.1 – effet changement QP sur débit, temp et qualité (Four-People).....	16
Tableau.2 - (BASKETBALL-DRILL-AI) effet changement QP sur débit, temp et qualité.....	17
Tableau.3 - (BASKETBALL DRILL-LD) effet changement QP sur débit, temp et qualité.....	17
Tableau.4 - (BASKETBALL DRILL-RA) effet changement QP sur débit, temp et qualité.....	18
Tableau.5 - (FOUR-PEOPLE-LD) effets des plusieurs pertes (1%, 5%, 10%, 20%) pour différents QP.....	19
Tableau.6 - (BASKETBALL-DRILL-AI) effets de pertes de 3% et 20% sur la qualité.....	22
Tableau.7 - (BASKETBALL-DRILL-LD) effets de pertes de 3% et 20% sur la qualité.....	22
Tableau.8 - (BASKETBALL-DRILL-RA) effets de pertes de 3% et 20% sur la qualité.....	22
Tableau.9 - (BASKETBALL-DRILL-AI-RA-LD) effets de pertes de 3%.....	24
Tableau.10 - (BASKETBALL-DRILL-AI-RA-LD) effets de pertes de 20%.....	26
Tableau.11 - (BASKETBALL-DRILL-AI-RA-LD) : l'impact de changement du QP sur la Qualité pour les deux PLR (3 et 20%).....	29

Listes des figures :

FIG.1 : comparaison entre la compression d'image avec perte et sans perte.....	3
FIG.2 : système de compression générique.....	4
FIG.3 : Chronologie de création des standards.....	5
FIG.4 : processus d'encodage et décodage de H.264.....	6
FIG.5 : Exemple de compression d'un vidéo avec un QP 22.....	10
FIG.6 : Exemple de fichier trace d'une vidéo (LD).....	12
FIG.7 : Exemple de fichier contient des paquets d'adresse IP émis.....	12
FIG.8 : FOUR-PEOPLE (PSNR en fonction de QP).....	15
FIG.9 : FOUR-PEOPLE (SSIM en fonction de QP).....	15
FIG.10 : Basketball-drill (PSNR en fonction de QP).....	16
FIG.11 : Basketball-drill (PSNR en fonction de QP).....	17
FIG.12 : FOUR-PEOPLE - Impact de PLR (PSNR en fonction de QP).....	18
FIG.13 : FOUR-PEOPLE - Impact de PLR (SSIM en fonction de QP).....	19
FIG.14 : (Basketball-drill-AI) - Impact de PLR (SSIM en fonction de QP).....	20
FIG.15 : (Basketball-drill-AI) - Impact de PLR (PSNR en fonction de QP).....	20
FIG.16 : (Basketball-drill-LD) - Impact de changement de PLR (PSNR en fonction de QP).....	21
FIG.17 : (Basketball-drill-LD) - Impact de changement de PLR (SSIM en fonction de QP).....	21
FIG.18 : (Basketball-drill-RA) - Impact de changement de PLR (PSNR en fonction de QP).....	21
FIG.19 : (Basketball-drill-RA) - Impact de changement de PLR (SSIM en fonction de QP).....	21

FIG.20 : (Basketball-drill) -Effet de changement de QP pour un PLR 3% (PSNR en fonction de QP).....	23
FIG.21 : (Basketball-drill) -Effet de changement de QP pour un PLR 3% (SSIM en fonction de QP).....	24
FIG.22 : (Basketball-drill) -Effet de changement de QP pour un PLR de 20% (PSNR en fonction de QP).....	25
FIG.23 : (Basketball-drill) -Effet de changement de QP pour un PLR de 20% (SSIM en fonction de QP).....	26
FIG.24 : (Basketball-drill-AI) -Impact de PLR pour les quatre QP (PSNR en fonction de PLR).....	27
FIG.25 : (Basketball-drill-AI) -Impact de PLR pour les quatre QP (SSIM en fonction de PLR).....	27
FIG.26 : (Basketball-drill-RA) -Impact de PLR pour les quatre QP (PSNR en fonction de PLR).....	28
FIG.27 : (Basketball-drill-RA) -Impact de PLR pour les quatre QP (SSIM en fonction de PLR).....	28
FIG.28 : (Basketball-drill-LD) -Impact de PLR pour les quatre QP (PSNR en fonction de PLR).....	28
FIG.29 : (Basketball-drill-LD) -Impact de PLR pour les quatre QP (SSIM en fonction de PLR).....	28

Sommaire :

Chapitre 1 : Généralités sur la compression des données

1	Introduction :	2
1.1	Compression des données :	2
1.1.1	Définition :	2
1.1.2	Types de compression :	2
1.1.2.1	Compression sans pertes :	2
1.1.2.2	Compression avec pertes :	3
1.2	Compression d'image numérique :	3
1.3	Compression de la vidéo numérique :	3
1.4	Quelque standards vidéo :	4
1.4.1	Chronologie :	4
1.4.2	Le standard H.264/ MPEG-4 AVC/MPEG4 PART 10 :	5
1.4.2.1	Définition :	5
1.4.2.2	Principes de fonctionnement de l'encodeur H.264:	5
1.4.2.3	Le but d'utiliser H.264 :	6
1.4.2.4	Domaines d'applications :	6
1.4.3	Le standard H.265/HEVC :	6
1.4.4	Comparaisons entre H.264 et H.265 :	7
1.5	Conclusion :	7

Chapitre 2 : Etude de JM, EvalVid et les métriques de mesure de qualité

2	Introduction :	8
2.1	Les métriques de qualité (PSNR et SSIM) :	8
2.1.1	Définition de PSNR :	8
2.1.2	Définition du SSIM :	9
2.2	Comparaison entre PSNR et SSIM :	9
2.3	Principe de l'encodeur JM :	9
2.4	Principe de simulateur EvalVid :	11
2.5	Conclusion :	13

Chapitre 3 : Etude d'effet de QP sur les trois configurations avant et après transmissions

3	Introduction.....	14
3.1	Avant transmission :.....	14
3.1.1	Etude de l'impact du changement du QP sur la qualité :	14
3.1.2	Etude de l'impact du changement du QP sur les trois configurations :	16
3.2	Après transmission :	18
3.2.1	Etude de l'impact de PLR sur la Qualité :.....	18
3.2.2	Etude de l'impact de changement du PLR sur la Qualité pour les trois configurations :	20
3.2.3	Effet de changement de QP :.....	23
3.2.3.1	Effet de changement de QP sur la qualité pour un PLR de 3% :.....	23
3.2.3.2	Effet de changement de QP sur la qualité pour un PLR de 20% :.....	25
3.2.4	Etude de l'impact de changement du QP sur la Qualité pour les trois configurations pour différents PLR :	27
3.3	Conclusion :.....	29

Introduction générale

De nos jours, le domaine des multimédias tel que l'imagerie ou la vidéo a bouleversé notre mode de vie, il connaît un développement très rapide et important. Les applications utilisant ou susceptibles d'utiliser le multimédia sont très diverses comme la médecine, l'enseignement, la vidéo surveillance, le control de trafic...etc.

Dans un autre côté, le volume des données transmises devient de plus en plus important ce qui engendre une complexité de traitement et un temps de transmission plus long, alors une compression des données est nécessaire. Cette dernière est l'objet de notre étude surtout sur la qualité de la vidéo avant et après la transmission.

Les vidéos sont très riches en termes d'informations ce qui requièrent une grande bande passante pour les transférer, cela pose un grand problème et des contraintes lorsqu'ils sont transmis sur les réseaux.

L'objectif de cette compression est de minimiser la taille de fichier pour optimiser l'usage des canaux de communication et aussi gagner en matière de temp, cette compression de vidéos sera faite en utilisant des encodeurs (dans notre cas nous allons utiliser le JM19).

Ce manuscrit est organisé comme suit :

Dans le premier chapitre, nous présenterons une étude générale sur la compression des images et des vidéos, les types des compressions et nous allons étudier les deux normes les plus récentes de compression (H.264, H.265) et finalement une petite comparaison entre les deux.

L'étude des calculateurs de qualités, l'encodeur JM et le simulateur EvalVid fait l'objet du deuxième chapitre, Dans un premier temps nous expliquerons le PSNR et SSIM et nous citerons les formules utilisées pour les calculs de ces deux métriques. Nous considérons ensuite une comparaison entre les deux, ensuite nous procédons à une étude d'une manière générale sur le principe de l'encodeur JM et le simulateur EvalVid. Nous introduirons d'abord l'encodeur JM (Joint Model) développé par l'équipe JVT (Joint Video Team), ces configurations et ces résultats, après nous verrons c'est quoi un simulateur EvalVid et comment ça marche.

Dans le dernier chapitre de ce travaille, nous allons évaluer l'effet de la compression sur la qualité des vidéos et l'effet des transmissions avec différents taux de pertes (3%, 5%, 10%, 20%) pour les trois configurations (AI, LD, RA), commençant par l'impact du changement du QP sur la qualité après la compression, ensuite nous discuterons et analyserons l'impact de PLR sur la qualité des vidéos.

En conclusion générale nous résumerons les résultats essentiels qui ont été obtenus et nous proposerons quelques perspectives.

Chapitre 1 :

Généralités sur la compression des données

1 Introduction :

Dans nos jours, la compression de donnée est nécessaire pour minimiser la taille des fichiers soit images, vidéos ou audio pour bénéficier en matière de stockage et de temps de transmission.

Dans ce chapitre, nous étudierons la compression des données (image et vidéo) et verrons les normes de compression de manière générale.

1.1 Compression des données :

1.1.1 Définition :

La Compression des données, comme nous pouvons l'appeler aussi le codage de source est l'opération informatique qui va nous permettre de réduire la taille des données, où les algorithmes de compression sont spécifiques à certains types de médias (images, vidéo, audio) [1] [4]

1.1.2 Types de compression :

1.1.2.1 Compression sans pertes :

Dans la compression sans perte, aucune information n'est perdue c'est-à-dire une récupération des données à 100% d'origine. Elle été utilisé pour les archives l'imagerie médicale, et les dessins techniques.

Exemples de méthodes sans perte [2] :

- DPCM et codage prédictif
- Encodage entropique
- Codes de chaîne
- DEFLATE

1.1.2.2 Compression avec pertes :

Dans la compression avec perte, l'information perdue est irréversible, elle a été adaptée aux images naturelles où les pertes sont acceptables car les pertes négligeables ne sont pas visuelles, cette technique utilise plusieurs méthodes comme [2] [4] :

- Par prédiction, par exemple l'ADPCM ;
- Par transformation (transformer le codage)
- La compression Fractale



FIG.1 : comparaison entre la compression d'image avec perte et sans perte.

1.2 Compression d'image numérique :

Compression d'image est un type de compression de données, elle a été utilisée pour minimiser la quantité de données sans dégrader beaucoup la qualité pour gagner en matière de stockage et bénéficier en transmission. Les formats les plus utilisés sont : JPEG, GIF et PNG. [2] [4] [5]

1.3 Compression de la vidéo numérique :

Une vidéo est une série successive d'images à un certain rythme, où l'œil humain croit que c'est une image animée. [5]

La compression vidéo est une autre méthode de compression de données, qui va réduire le débit par effacer les éléments inutiles, elle contient les deux processus suivants :

- **L'encodeur** : convertit le format de la vidéo au format compressé pour gagner en matière de stockage et bénéficier en transmission, il peut être sans perte ou avec perte, les formats vidéo populaires sont : MPEG, DIVX, AVI, WMV et MP4
- **Le décodeur** : c'est la technique inverse de l'encodeur où la vidéo va se revoir à l'état original (dans le cas sans pertes).

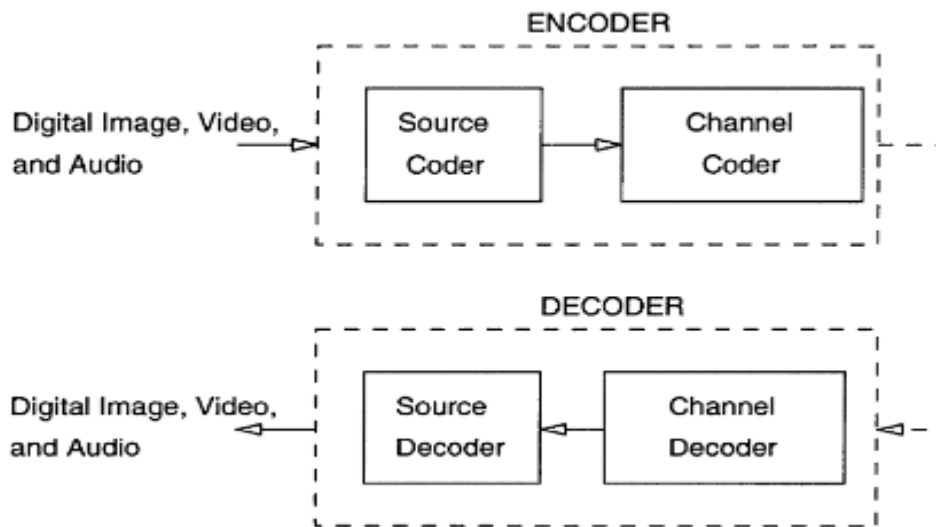


FIG.2 : système de compression générique

1.4 Quelques standards vidéo :

1.4.1 Chronologie :

Il existe 3 organismes de standardisation à une certaine chronologie montrée dans la figure [6] (FIG.3) :

- ISO International Standardisation Organizations [6]
- IEC International Électrotechnique Commission [6]
- ITU International Télécommunication Union [6]

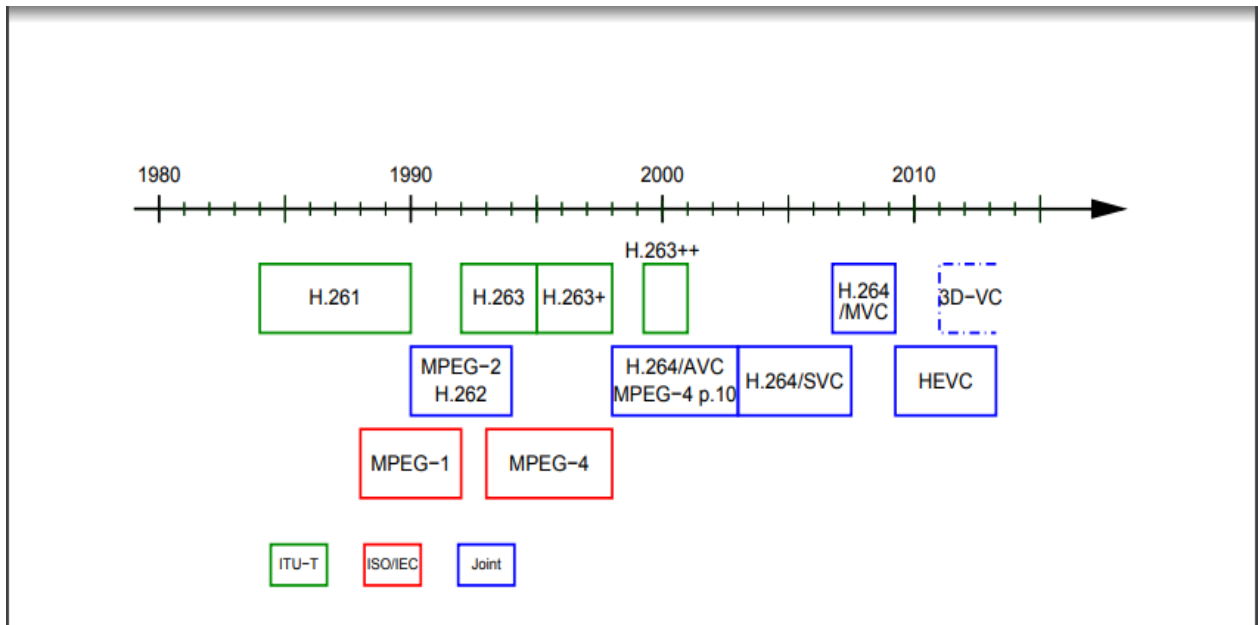


FIG.3 : Chronologie de création des standards

1.4.2 Le standard H.264/ MPEG-4 AVC/MPEG4 PART 10 :

1.4.2.1 Définition :

C'est une norme récente de codage (compression) vidéo développée par l'UIT-T et l'ISO/CEI (Joint Vidéo Team), elle été publié pour la première fois on 2003, elle s'appuie sur caractéristique des normes anciennes telles que h.262 et h.263. Cette norme est basée sur la compression spatiale et temporelle, elle offre une meilleure efficacité et flexibilité en compression et aussi en transmission et stockage vidéo. [7]

1.4.2.2 Principes de fonctionnement de l'encodeur H.264:

L'encodeur H.264 exécute plusieurs processus sur la vidéo comme la prédiction, la transformation ensuite le codage, le résultat est un flux binaire, pour le décodage il utilise des processus inverse comme la transformation inverse ensuite la reconstruction pour avoir la vidéo décodée. (Tous les processus dans [7])

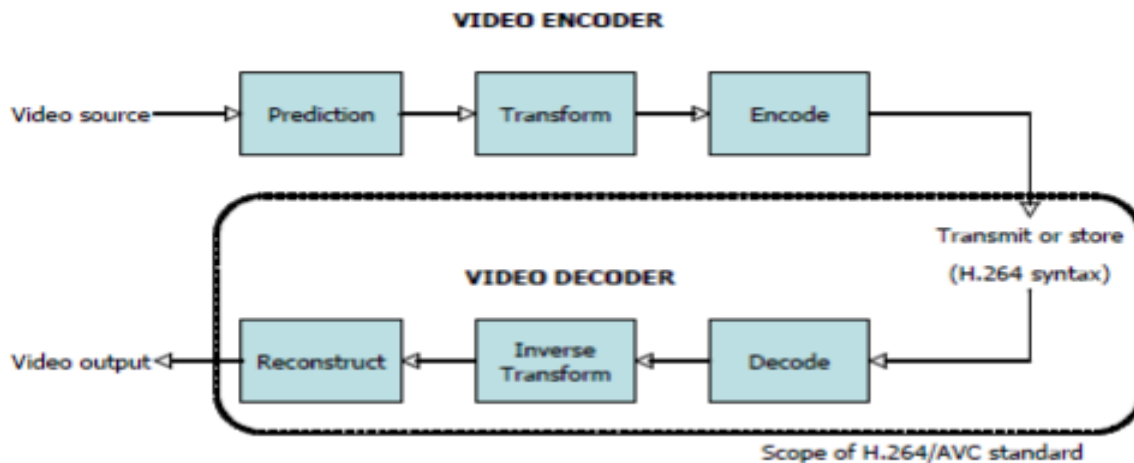


FIG.4 : processus d'encodage et décodage de H.264

1.4.2.3 Le but d'utiliser H.264 :

La H.264 a été créée pour [6] :

- Minimiser le taux de distorsion de performance
- Jusqu'à 60% de réduction du taux pour la même qualité par rapport au MPEG-2

1.4.2.4 Domaines d'applications :

La H.264 est adoptée pour plusieurs applications, notamment [7] :

- High Definition DVDs (HD-DVD and Blu-Ray formats)
- High Definition TV broadcasting in Europe
- Produits Apple, notamment téléchargements de vidéos iTunes, iPod vidéo et MacOs
- NATO and US DoD video applications
- Mobile TV broadcasting
- Internet video
- Videoconferencing

1.4.3 Le standard H.265/HEVC :

C'est la norme la plus récente de codage (compression) vidéo développée par l'UIT-T et l'ISO/CEI (Joint Vidéo Team), elle a été publiée pour la première fois en janvier 2013. [8]

[Toutes les normes de codage vidéo récentes exploitent la redondance temporelle par prédiction d'inter-image et exploitent la redondance spatiale en utilisant la prédiction intra-image en plus de transformer le processus de codage de cette prédiction]. [9]

La structure de conception de la norme HEVC n'a pas grande différence au norme antérieure H.264, tandis qu'elle a les mêmes étapes de compression de la norme h.264. [9]

La HEVC représente le signal vidéo couleur avec l'espace couleur de YCbCr en 4 :2 :0, la composante Y pour la luminance et Cb et Cr pour la chrominance de la vidéo. [9]

1.4.4 Comparaisons entre H.264 et H.265 :

HEVC est la norme la plus récent, elle a un grand succès par rapport au H.264, elle est adaptée pour des résolutions vidéo jusqu'au codage vidéo UHD, qui peut atteindre jusqu'à 8K (8192*4320) pixels alors que la H.264 peut atteindre jusqu'à 4K (4,096*2,304). [9]

Pour un même débit, la HEVC produit une qualité mieux que la H.264 et la HEVC promet d'énormes économies de bande passante d'environ 50% par rapport au H.264 avec la même qualité utilisée pour les deux. [9]

La fréquence d'image est limitée pour la H.264 à 59.94 (fps) alors que la HEVC peut supporter jusqu'à 300 (fps). [9]

1.5 Conclusion :

Dans ce chapitre nous avons vu c'est quoi une compression de données, ces méthodes (image, vidéo) et qu'il existe deux types de compression (avec pertes, sans pertes). Nous avons parlé des deux normes les plus récentes (H.264 et H.265), leurs principes de fonctionnement et l'intérêt de leurs existences. Enfin nous avons présenté une comparaison entre les deux normes. Nous avons conclu que HEVC est la meilleure et la plus efficace. Cependant, le HEVC souffre d'une très grande complexité algorithmique ce qui limite son utilisation dans les systèmes embarqués.

Le chapitre suivant fait l'objet d'une étude de l'encodeur JM et les métriques de qualité (PSNR et SSIM).

Chapitre 2 :

Etude de JM, EvalVid et les

2 Introduction :

Dans ce chapitre nous aborderons d'abord la définition des calculateurs de qualité PSNR et SSIM. Ensuite la comparaison entre PSNR et SSIM ainsi que la définition et le principe de fonctionnement de l'encodeur JM, et finalement nous définirons le simulateur EvalVid et nous présenterons son principe.

2.1 Les métriques de qualité (PSNR et SSIM) :

2.1.1 Définition de PSNR :

PSNR est un terme technique qui signifie le rapport de la puissance maximale possible d'un signal sur la puissance de corruption du bruit, le PSNR est généralement exprimé en décibels.[12]

Il est apparu comme processus comparable de la qualité des codages de signaux vidéo compressés, la valeur donnée est une mesure de la différence entre l'image d'origine et l'image compressée.

Le PSNR est généralement défini par l'erreur quadratique moyenne (MSE) par l'équation suivante [12] :

$$\begin{aligned} PSNR &= 10 \cdot \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \\ &= 20 \cdot \log_{10} \left(\frac{MAX_I}{\sqrt{MSE}} \right) \\ &= 20 \cdot \log_{10}(MAX_I) - 10 \cdot \log_{10}(MSE) \end{aligned}$$

Où MAX correspond à la valeur de pixel maximale possible de l'image (255) et MSE représente l'erreur quadratique moyenne, qui est la moyenne du carré des différences de pixels des deux images défini par [12] :

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i, j) - K(i, j)]^2$$

Étant donné une image monochrome $m \times n$ sans bruit I et son approximation bruyante K où $m \times n$ est la taille de l'image [12]

2.1.2 Définition du SSIM :

C'est une métrique qui a démontré un bon accord avec des observateurs humains où nous utilisons des images de référence, cette métrique analyse la différence de visualisation, SSIM a été utiliser pour prédire la performance humaine dans certains taches (images médicales radiologiques). [13]

Le SSIM évalue une image originale X par rapport à une référence image Y pour quantifier leur similarité visuelle, la forme générale de SSIM défini comme suit [13] :

$$SSIM(x,y)=[l(x,y)]^\alpha.[c(x,y)]^\beta.[r(x,y)]^\gamma$$

Où x et y sont des paires de fenêtres carrées locales, α et β et γ sont des paramètres qui définissent l'importance relative de chaque composant.

L'indice l (x, y) est lié aux différences de luminance, c (x, y) aux différences de contraste et r (x, y) aux variations de structure entre x et y. [13]

2.2 Comparaison entre PSNR et SSIM :

D'après le travail de AHMED SEGHIR Zianou, HACHOUF Fella, ARDJOUNE Mourad, FALEK Salem [14], on a vu que l'étude comparative entre SSIM et PSNR nous a conduit à conclu que la méthode SSIM donne des résultats satisfaisants par rapport aux autres méthodes.

2.3 Principe de l'encodeur JM :

JM est un logiciel référence H.264/AVC développé par Karsten.Suehring[10], la dernière version et que nous avons utilisé est JM 19, ce logiciel peut être utilisé sur Windows ou Linux, avec ce logiciel on peut avoir un vidéo référence avec un taux de compression que nous voulons (on va utiliser 22, 27, 32, 37) et nous pouvons avoir aussi des vidéos référence pour les 3 configurations (AI-RA-LD) depuis JM 19, pour démarrer l'encodage nous devons utiliser cette commande « **lencod.exe -d default.cfg** », le fichier de configuration (.cfg) contient plusieurs paramètres, nous citons les plus important :

- InputFile = "BasketballDrill_832x480_50.yuv", où on va donner la vidéo originale pour l'encoder.
- FramesToBeEncoded = 501, c'est le nombre d'image que la vidéo contient
- FrameRate = 50, c'est le FPS
- SourceWidth = 832 la taille de la vidéo originale
- SourceHeight = 480 Et on dit choisir la résolution
- OutputWidth = 832 De la vidéo référence.
- OutputHeight = 480

- QPISlice =22, la partie la plus importante où on doit choisir le taux de compression pour les différents types d'image (I, P, B) pour avoir un GOP spécifique.
- Donc il y a beaucoup de commandes, pour plus d'informations vous pouvez consulter la référence [11], finalement le JM nous donne une vidéo référence et quelque résultats (PSNR, MSE, temp de compression, débit).

```

00478< P > 119776 24 39.961 42.331 42.808 29988 9958 FRM 1
00479< P > 87600 25 39.638 42.152 42.617 28776 9638 FRM 1
00480< P > 168464 23 40.368 42.667 43.100 31991 10717 FRM 2
00481< P > 86048 25 39.747 42.369 42.713 29039 9952 FRM 1
00482< P > 115272 24 39.904 42.476 42.906 30214 10253 FRM 1
00483< P > 96560 25 39.638 42.279 42.626 29558 10360 FRM 1
00484< P > 160024 23 40.297 42.695 43.152 31685 10598 FRM 2
00485< P > 93264 25 39.663 42.334 42.666 29402 10136 FRM 1
00486< P > 130488 24 39.901 42.449 42.849 30336 10299 FRM 1
00487< P > 105128 25 39.582 42.208 42.582 29639 10403 FRM 1
00488< P > 164528 23 40.297 42.683 43.173 14384 3755 FRM 2
00489< P > 94608 25 39.682 42.273 42.682 13256 3666 FRM 1
00490< P > 121704 24 39.969 42.459 42.915 13973 3827 FRM 1
00491< P > 92688 25 39.596 42.227 42.642 13423 3759 FRM 1
00492< P > 156640 23 40.374 42.765 43.290 14340 3769 FRM 2
00493< P > 84024 25 39.780 42.379 42.776 13162 3551 FRM 1
00494< P > 115864 24 39.932 42.532 42.949 13727 3679 FRM 1
00495< P > 90384 25 39.703 42.356 42.727 29553 10097 FRM 1
00496< P > 154504 23 40.330 42.813 43.281 31929 10721 FRM 2
00497< P > 86256 25 39.831 42.450 42.811 29034 9984 FRM 1
00498< P > 118208 24 40.006 42.547 42.953 13734 3797 FRM 1
00499< P > 92816 25 39.726 42.385 42.725 13257 3650 FRM 1
Inserting 486/1458 cabac_zero_word syntax elements/bytes <Clause 7.4.2.10>
Inserting 481/1441 cabac_zero_word syntax elements/bytes <Clause 7.4.2.10>
00500< P > 393944 23 38.847 98.124 98.124 39234 20865 FRM 2
-----
Total Frames: 501
LeakyBucketRate File does not exist. Using rate calculated from avg. rate
Number Leaky Buckets: 8
Rmin Bmin Fmin
5997250 914456 546136
7496550 546136 546136
8995850 546136 546136
10495150 546136 546136
11994450 546136 546136
13493750 546136 546136
14993050 546136 546136
16492350 546136 546136
-----
Average data all frames
-----
Total encoding time for the seq. : 25550.564 sec <0.02 fps>
Total ME time for sequence : 9388.002 sec
Y < PSNR <dB>, cSNR <dB>, MSE > : < 39.972, 39.961, 6.56125 >
U < PSNR <dB>, cSNR <dB>, MSE > : < 42.659, 42.549, 3.61557 >
V < PSNR <dB>, cSNR <dB>, MSE > : < 43.117, 43.004, 3.25610 >
Total bits : 60093160 <I 546136, P 59546696, NUB 328>
Bit rate <kbit/s> @ 50.00 Hz : 5997.32
Bits to avoid Startcode Emulation : 15104
Bits for parameter sets : 328
Bits for filler data : 0
-----
Exit JM 19 <FRExt> encoder ver 19.0
C:\Users\seif eddine\Desktop\master2encodeur\JM_Test\video2 LD.AI.RA<basketball

```

FIG.5 : Exemple de compression d'un vidéo avec un QP 22

Nous remarquons depuis FIG.5 que JM nous donne des informations sur le débit, le temp de compression et il nous donne aussi le PSNR mais pas un résultat final alors nous devons le calculer par la formule suivante :

$$\text{PSNR} = (6*y+u+v) / 8$$

2.4 Principe de simulateur EvalVid :

EvalVid est un simulateur qui contient plusieurs outils permettant d'évaluer la qualité de la vidéo qui va être transmis sur un réseau simulé par (EvalVid ou un autre simulateur réseau).

Il est utilisé généralement par les chercheurs qui veulent évaluer un réseau, il peut être utilisé avec certain format de vidéo (H.263 /H.264 /MPEG4). [17]

EvalVid contient certains paramètres tels que les taux de perte, les retards et la gigue que nous pouvons appliquer sur la vidéo, il contient aussi des métriques de qualités (PSNR et SSIM).

Pour utiliser EvalVid comme outil d'envoi et de réception, comme un outil d'évaluation nous devons suivre les étapes suivantes :

1. Partie avant transmission :

- Choisir une vidéo originale que nous allons faire une expérience sur elle.
- Passer la vidéo par un encodeur (nous avons utilisé JM 19) pour avoir une vidéo référence avec les paramètres que vous voulez et un fichier de bitstream.
- Nous allons utiliser un conteneur MP4 pour contenir le bitstream par la commande suivante :

```
MP4Box -hint -mtu 1024 -fps 30 -add a03.264 a03.mp4
```

Où MP4box est une application, -fps 30 est le nombre d'image par seconde, a03.264 le bitstream, a03.mp4 est le conteneur MP4.

2. Partie transmission et réception :

Cette partie est composé de deux autres parties :

➤ Partie en ligne :

- On doit d'abord extraire un fichier SDP et lui configurer depuis le conteneur MP4 par la commande suivante :

```
MP4Box -std -sdp a01.mp4 > a01.sdp
```

- Nous devrions créer un fichier de trace et envoyer le fichier conteneur au destinataire à l'aide de la commande suivante :

```
mp4trace -f -s 192.168.0.2 12346 a01.mp4 > st_a01
```

Où mp4trace est une application, 192.168.0.2 l'adresse de récepteur, 12346 le port de destination, a01.mp4 fichier container, st_a01 le fichier trace

1	H	81497	80	0.033
2	H	3869	4	0.034
3	H	5254	6	0.034
4	H	4328	5	0.050
5	H	9654	10	0.067
6	H	4069	4	0.084
7	H	6129	6	0.099
8	H	4479	5	0.116
9	H	10946	11	0.132
10	H	4174	5	0.149
11	H	6224	7	0.166
12	H	4636	5	0.182
13	H	10600	11	0.199
14	H	4461	5	0.215
15	H	6161	7	0.231
16	H	4363	5	0.248
17	H	10474	11	0.264
18	H	4162	5	0.281
19	H	5907	6	0.297
20	H	4457	5	0.314
21	H	10107	10	0.330
22	H	3946	4	0.348
23	H	5811	6	0.364
24	H	4680	5	0.380
25	H	14184	14	0.397
26	H	4762	5	0.413
27	H	7427	8	0.429
28	H	4890	5	0.447
29	H	13530	14	0.467
30	H	5398	6	0.479

FIG.6 : Exemple de fichier trace d'une vidéo (LD)

- Nous devons utiliser une application (TCPdump) pour suivre les paquets IP de l'émetteur et le récepteur en utilisant les commandes suivantes :

Tcpdump -n -tt -v Udp port 12346 > sd_a01

Utiliser pour l'émetteur où Udp est le Protocol utiliser, sd_a01 est le fichier qui contient les paquets émis.

Tcpdump -n -tt -v Udp port 12346 > rd_a01

Utiliser pour le récepteur où rd_a01 est le fichier qui contient les paquets reçus

```

1554573171.189173 IP (tos 0x0, ttl 64, id 3350, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189186 IP (tos 0x0, ttl 64, id 3351, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189190 IP (tos 0x0, ttl 64, id 3352, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189194 IP (tos 0x0, ttl 64, id 3353, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189198 IP (tos 0x0, ttl 64, id 3354, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189201 IP (tos 0x0, ttl 64, id 3355, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189205 IP (tos 0x0, ttl 64, id 3356, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189209 IP (tos 0x0, ttl 64, id 3357, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189213 IP (tos 0x0, ttl 64, id 3358, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024
1554573171.189217 IP (tos 0x0, ttl 64, id 3359, offset 0, flags
[DF], proto UDP (17), length 1052)
 127.0.0.1.59654 > 127.0.0.1.12346: UDP, length 1024

```

FIG.7 : Exemple de fichier contient des paquets d'adresse IP émis

➤ Partie hors ligne :

- Nous allons faire la même chose comme la partie en ligne mais nous ne devons pas utiliser un fichier SDP.

- On utilise une commande pour l'écoute hors ligne :

Netcat -l -u -p 12346 > /dev/null

- Nous devons utiliser les commande Tcpdump et Mp4trace comme nous l'avons fait précédemment mais cette fois-ci nous faisons un peu de changement sur la commande Mp4trace comme suit :

mp4trace -f -s 127.0.0.1 12346 a01.mp4 > st_a01

Où 127.0.0.1 est l'adresse de la boucle locale.

3. Partie d'évaluation :

- Nous devons d'abord corrompu le fichier en ajoutant un délai par la commande :

awk '{print \$3}' delay_a01.txt | hist - 0 .05 50.

Ou en ajoutant un bruit blanc :

eg sd_a01 rd_a01g st_a01 AWGN 250000.

Ou manuelle par effacer un certain nombre de paquets IP pour le fichier rd_01

- Après nous calculons la différence entre la vidéo référencée et la vidéo corrompue, en utilisant la commande suivante :

Psnr 352 288 420 akiyo_cif.yuv a01e.yuv > psnr_a01e.txt

Où 352 et 288 sont les dimensions de la vidéo ,420 Sous-échantillonnage de chrominance, akiyo_cif.yuv la vidéo original, a01e.yuv la vidéo corrompue, psnr_a01e.txt le résultat de calcul de la moyenne de PSNR.

2.5 Conclusion :

Nous avons vu dans ce chapitre les métriques de distorsion (PSNR, SSIM) utilisées pour calculer la différence de qualité entre deux vidéos, une comparaison entre les deux calculateurs, le principe de l'encodeur JM19 et enfin nous avons étudié le simulateur EvalVid.

Le chapitre suivant fait l'objet d'une étude pratique réalisée par le simulateur EvalVid et l'encodeur JM19 pour voir l'effet du facteur de qualité (QP) et le taux de perte de paquets (PLR) sur la qualité des vidéos.

Chapitre 3 :

Etude de l'effet de QP sur les trois configurations avant et après transmissions

3 Introduction

Dans ce chapitre, nous nous intéressons à l'étude de l'impact de la compression avec différents QP sur la qualité de la vidéo avant et après la transmission.

Nous utilisons le PSNR et le SSIM comme outils pour calculer le changement de la qualité des deux vidéos (Basketball-drill et Four-people) qui nous permettra de voir l'impact des différents changements que nous effectuerons.

Pour appliquer cette étude, nous utiliserons trois configurations qui sont Random Access (RA), All intra (AI) , Low Delay (LD) sur la vidéo (Basketball-drill) et la configuration LD seulement pour la vidéo (Four-people).

Ensuite, nous étudierons l'impact du QP sur la qualité de la vidéo pour les trois configurations avant transmission, puis nous étudierons la différence entre le PSNR de MATLAB et JM.

Pour la partie après transmission, nous étudierons l'impact du taux de perte de paquets (PLR) sur la qualité de la vidéo et l'effet du changement du QP. Pour le PLR nous utiliserons un type de perte BURST.

On finit par une petite conclusion qui résume et qui englobe ce qu'on a traité dans ce chapitre.

3.1 Avant transmission :

3.1.1 Etude de l'impact du changement du QP sur la qualité :

FIG.8-9 : représentent le changement la qualité vidéo (référence SSIM et PSNR) en fonction du QP, pour la vidéo Four-People, nous observons à chaque fois que le QP augmente, la qualité diminue.

On remarque aussi d'après **Tableau.1** que le débit et le temp va être diminuer qu'à chaque fois on va augmenter le QP.

Exemple : si nous choisissons des valeurs différente QP22 et QP32 où on peut voir clairement que PSNR et SSIM diminuent (voir les valeurs exactes ce référé au Tableau.1 - (Four-People))

Car à chaque fois que nous augmentant le QP nous provoquant plus de compression sur la vidéo qui induit des pertes de données résultant une chute du PSNR et du SSIM, de débit et du temps.

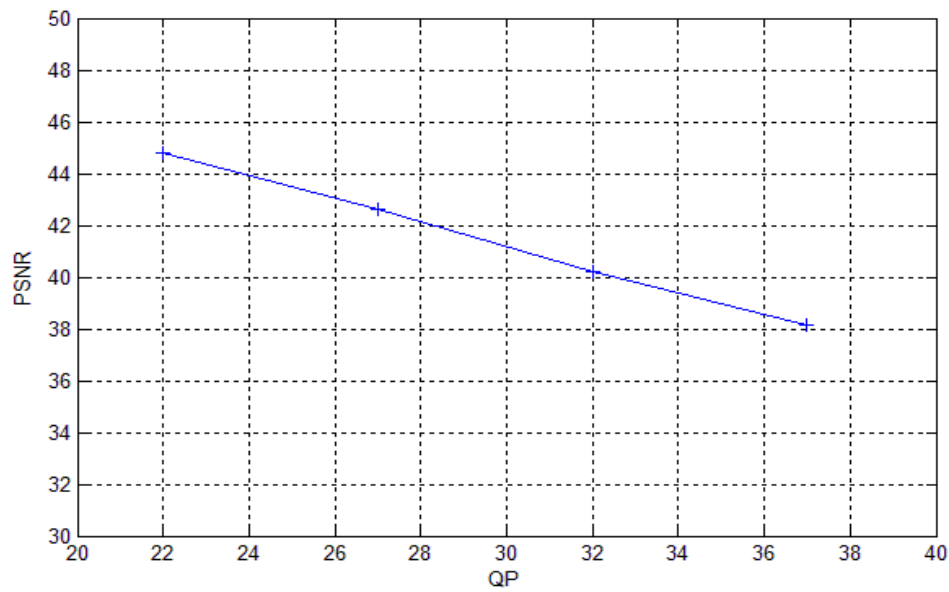


FIG.8 : FOUR-PEOPLE (PSNR en fonction de QP)

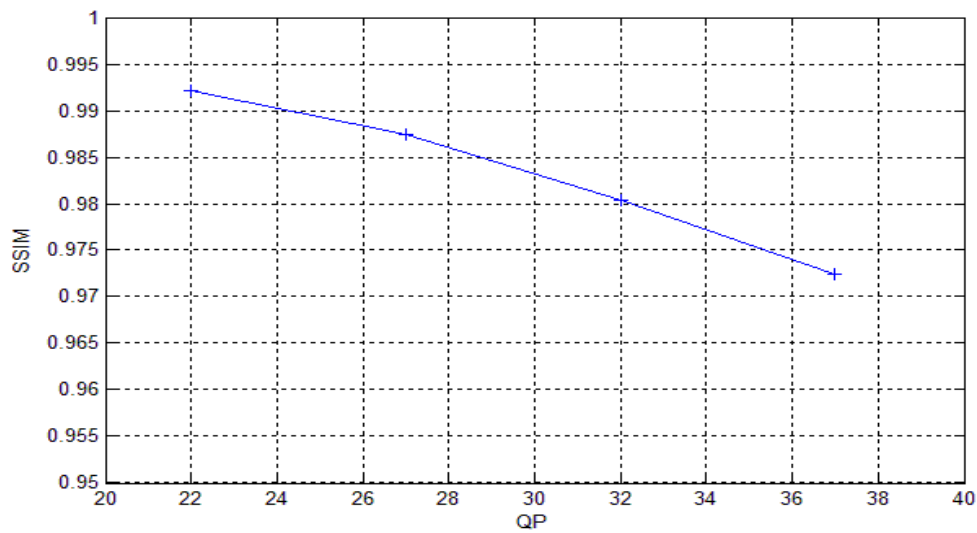


FIG.9 : FOUR-PEOPLE (SSIM en fonction de QP)

Tableau.1 - effet changement QP sur débit, temp et qualité (Four-People)

QP	22	27	32	37
Bit rate (Kb/s)	3616.85	1312.33	661.86	405.64
Temp (S)	23276.976	21357.286	20572	13933.515
SSIM	0.9922	0.9874	0.9804	0.9724
PSNR(Matlab)	44.8004	42.6163	40.2051	38.1364

3.1.2 Etude de l'impact du changement du QP sur les trois configurations :

FIG.10-11 : les deux figures représentent le changement de la qualité vidéo (référence SSIM et PSNR) en fonction du QP, on remarque que pour les trois configurations on obtient presque les mêmes résultats c'est-à-dire le changement de QP va affect les trois configurations en même façon.

Exemple : on choisit le QP22 et QP37 pour le cas d'une vidéo référence AI le PSNR continue de diminuer de 43.0618DB jusqu'à 34.4361DB on obtient alors une différence de 8.6257DB.

Et pour une vidéo référence LD la différence est de 7.9498DB et enfin pour une vidéo référence RA la différence est 8.0891DB

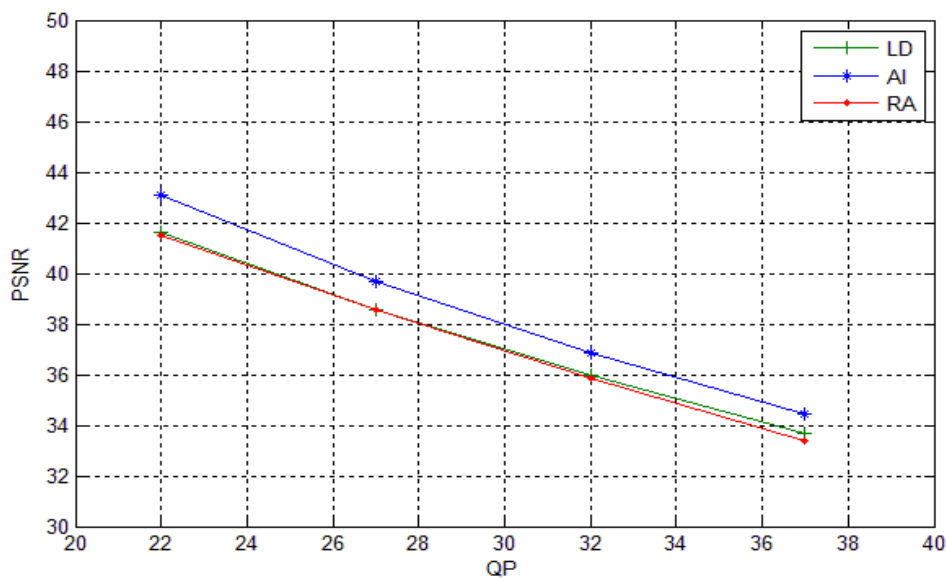


FIG.10 : Basketball-drill (PSNR en fonction de QP)

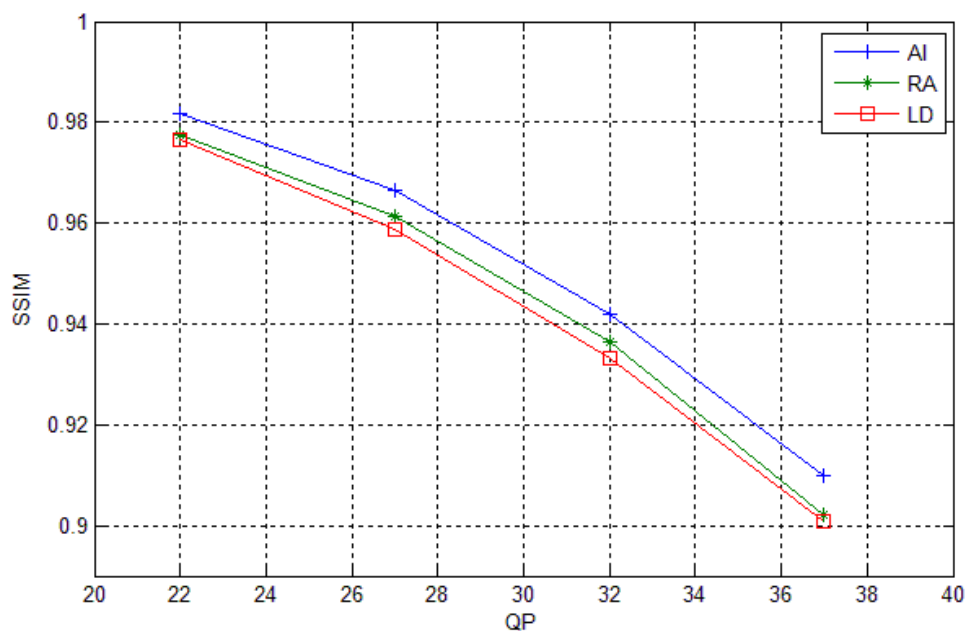


FIG.11 : Basketball-drill (PSNR en fonction de QP)

Tableau.2 - (BASKETBALL-DRILL-AI) effet changement QP sur débit, temp et qualité :

QP	22	27	32	37
Bit rate (Kb/s)	27133.14	15113.67	8128.02	4490
Temp(S)	6732.228	5983.210	5388.14	4989.423
PSNR(Matlab)	43.0618	39.6954	36.8485	34.4361
SSIM	0.9817	0.9664	0.9418	0.9099

Tableau.3 - (BASKETBALL DRILL-LD) effet changement QP sur débit, temp et qualité :

QP	22	27	32	37
Bit rate (Kb/s)	5997.32	2959.29	1469.65	741.69
Temp(S)	25550.564	24622.7	24196.708	23210.268
PSNR(Matlab)	41.6203	38.5869	35.9968	33.6705
SSIM	0.9764	0.9587	0.9333	0.9010

Tableau.4 - (BASKETBALL DRILL-RA) effet changement QP sur débit, temp et qualité :

QP	22	27	32	37
Bit rate (Kb/s)	4378.09	2157.49	1092.31	593.73
Temp(S)	37633.112	37646.568	36707.812	35511.776
PSNR(Matlab)	41.4954	38.5486	35.8306	33.4063
SSIM	0.9775	0.9614	0.9365	0.9021

3.2 Après transmission :

3.2.1 Etude de l'impact de PLR sur la Qualité :

FIG.12-13 : représentent les changements de la qualité en fonction du QP après une application des différents PLR (3%,5%,10%,20%) pour une vidéo référence (Four-people).

On remarque qu'à chaque fois le PLR augmente pour les quatre QP (22,27,32,37), nous observons une perte de qualité de la vidéo (PSNR et SSIM diminue) c'est-à-dire qu'à chaque fois que nous procédant à une transmission de vidéo on va perdre des paquets ce qui induit une perte de qualité.

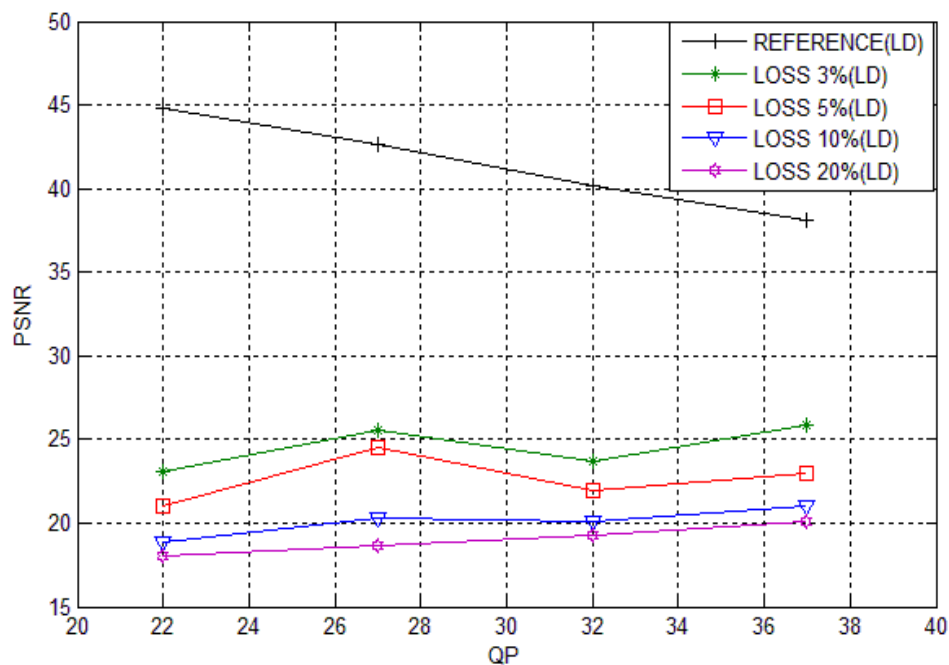


FIG.12 : FOUR-PEOPLE - Impact de PLR (PSNR en fonction de QP)

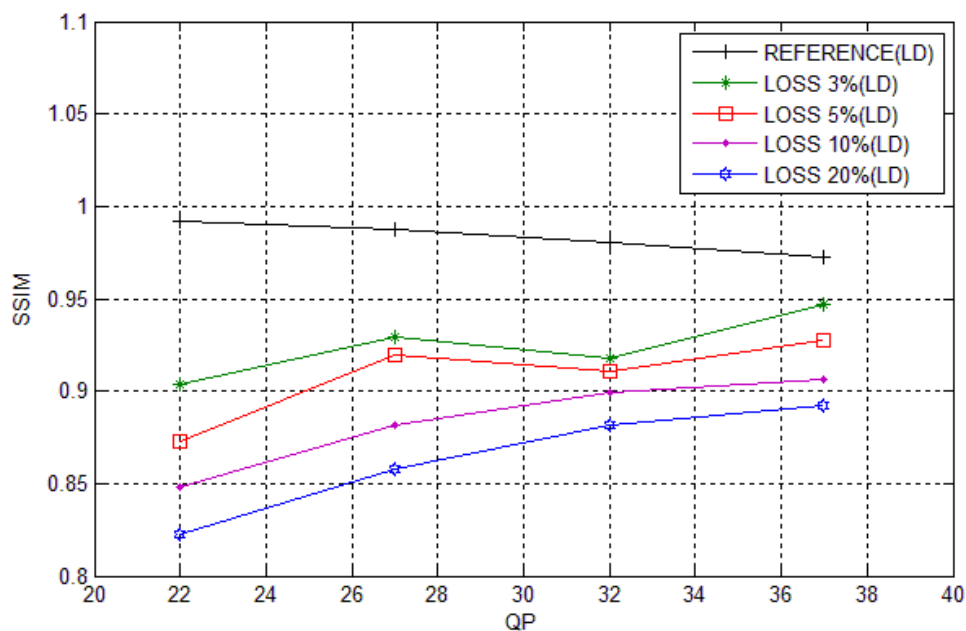


FIG.13 : FOUR-PEOPLE - Impact de PLR (SSIM en fonction de QP)

Tableau.5 - (FOUR-PEOPLE-LD) effets des plusieurs pertes (1%, 5%, 10%, 20%) pour différents QP :

QP	22	27	32	37
PSNR (référence)	44.8004	42.6163	40.2051	38.1364
SSIM (référence)	0.9922	0.9874	0.9804	0.9724
PSNR (loss 3%)	23.0912	25.6010	23.6736	25.8790
SSIM (loss 3%)	0.9036	0.9289	0.9182	0.9471
PSNR (loss 5%)	20.9963	24.4943	21.9919	23.0237
SSIM (loss 5%)	0.8728	0.9200	0.9104	0.9271
PSNR (loss 10%)	18.8253	20.2633	20.1386	20.9977
SSIM (loss 10%)	0.8481	0.8812	0.9092	0.9060
PSNR (loss 20%)	18.0605	18.6321	19.2669	20.0907
SSIM (loss 20%)	0.8228	0.8577	0.8813	0.8918

3.2.2 Etude de l'impact de changement du PLR sur la Qualité pour les trois configurations :

FIG.-14-15-16-17-18-19 : Représentent l'effet de changement de PLR sur la qualité de la vidéo pour différents QP par rapport à trois vidéo référence AI-RA-LD.

On utilise les deux PLR (3% et 20%) on remarque qu'à chaque fois que le QP augmente la qualité diminue mais pas de la même façon, on observe que pour un QP22 nous allons perdre beaucoup plus de qualité et à chaque fois que nous augmentons le QP nous perdons moins de qualité par rapport au QP22, dans une vidéo référence de QP22 on constate un nombre de paquets plus grand que pour les autres QP et à chaque fois que le QP augmente on obtient moins de paquets, c'est-à-dire si on applique une perte de 3% pour les deux vidéos l'une de QP22 et l'autre de QP37, la première vidéo va perdre plus de paquets car ils contiennent un nombre de paquets plus grand qu'une vidéo de QP37.

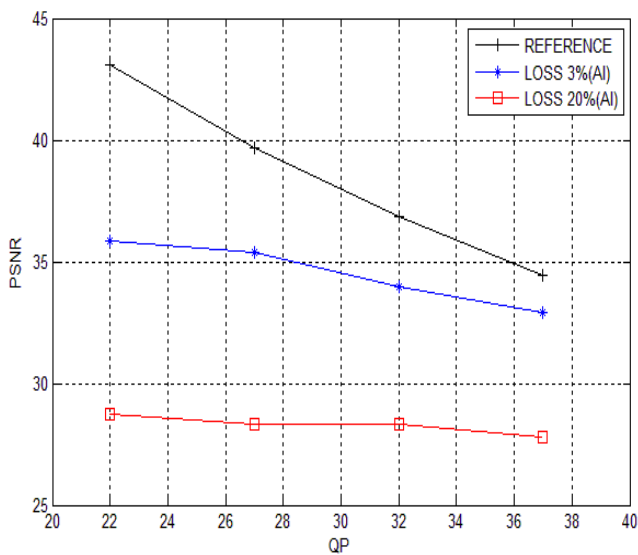


FIG.14 : (Basketball-drill-AI) -
Impact de PLR
(PSNR en fonction de QP)

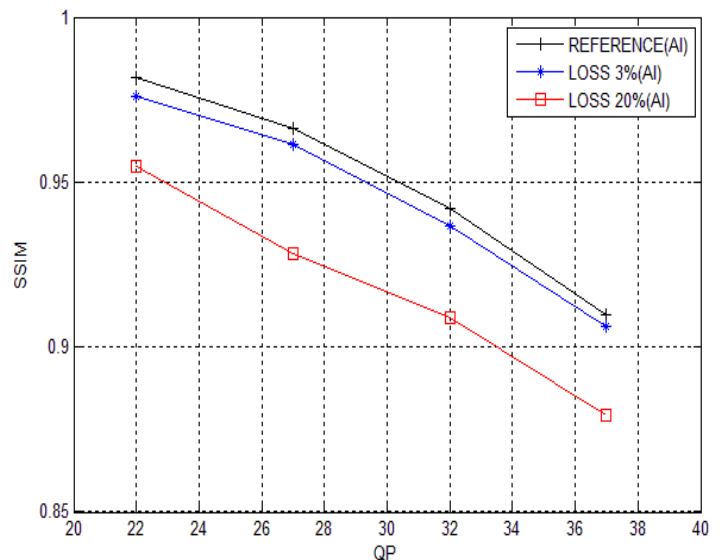
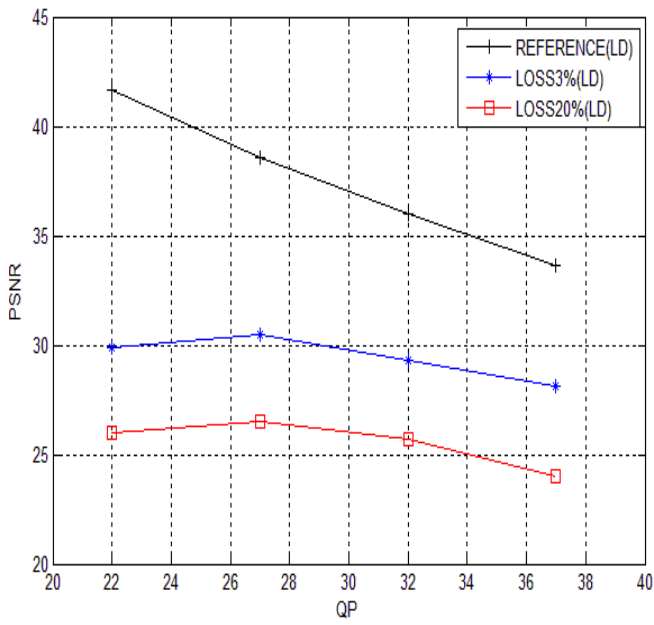
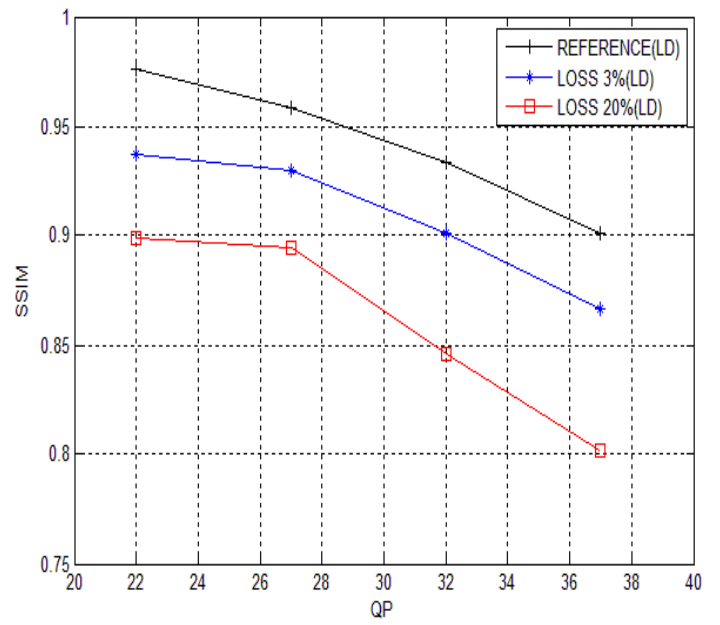


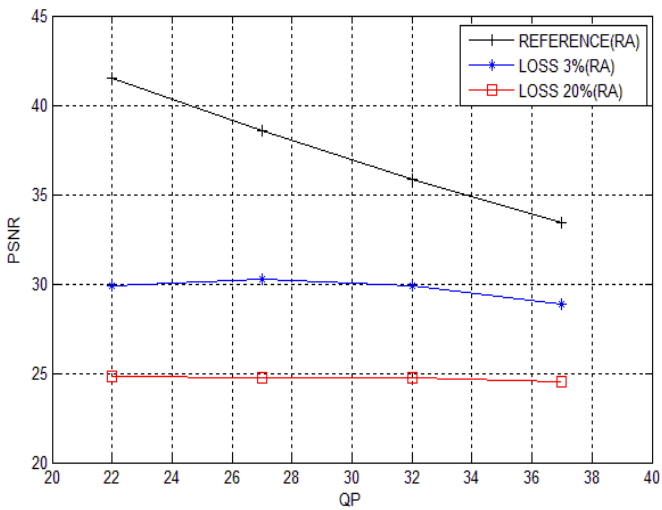
FIG.15 : (Basketball-drill-AI) -
Impact de PLR
(SSIM en fonction de QP)



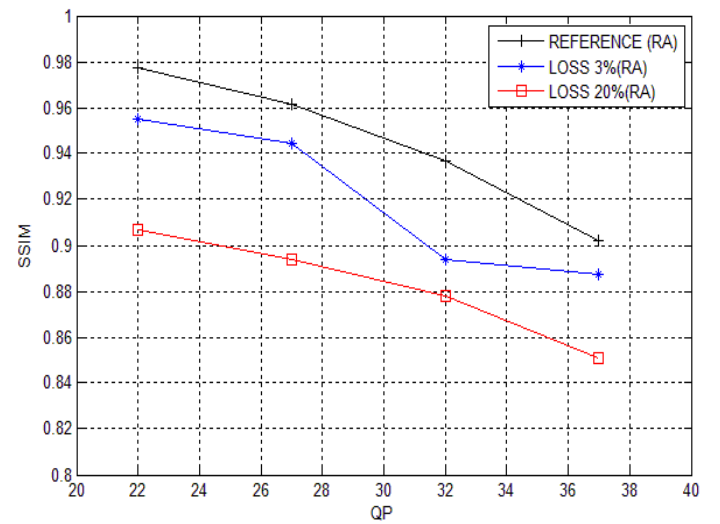
**FIG.16 : (Basketball-drill-LD) -
Impact de changement de PLR
(PSNR en fonction de QP)**



**FIG.17 : (Basketball-drill-LD) -
Impact de changement de PLR
(SSIM en fonction de QP)**



**FIG.18 : (Basketball-drill-RA)-
Impact de changement de PLR
(PSNR en fonction de QP)**



**FIG.19 : (Basketball-drill-RA)-
Impact de changement de PLR
(SSIM en fonction de QP)**

Tableau.6 - (BASKETBALL-DRILL-AI) effets de pertes de 3% et 20% sur la qualité :

QP	22	27	32	37
PSNR (référence)	43.0618	39.6954	36.8485	34.4361
SSIM (référence)	0.9817	0.9664	0.9418	0.9099
PSNR (loss 3%)	35.8495	35.3716	33.9640	32.9138
SSIM (loss 3%)	0.9759	0.9614	0.9367	0.9063
PSNR (loss 20%)	28.7252	28.3144	28.3131	27.8234
SSIM (loss 20%)	0.9549	0.9282	0.9089	0.8792

Tableau.7 - (BASKETBALL-DRILL-LD) effets de pertes de 3% et 20% sur la qualité :

QP	22	27	32	37
PSNR (référence)	41.6203	38.5869	35.9968	33.6705
SSIM (référence)	0.9764	0.9587	0.9333	0.9010
PSNR (loss 3%)	29.9144	30.4580	29.3330	28.0933
SSIM (loss 3%)	0.9372	0.9295	0.9008	0.8767
PSNR (loss 20%)	25.9860	26.5013	25.7329	24.0200
SSIM (loss 20%)	0.8989	0.8944	0.8462	0.8018

Tableau.8 - (BASKETBALL-DRILL-RA) effets de pertes de 3% et 20% sur la qualité :

QP	22	27	32	37
PSNR (référence)	41.6203	38.5869	35.9968	33.6705
SSIM (référence)	0.9764	0.9587	0.9333	0.9010
PSNR (loss 3%)	29.8809	30.24306	29.8920	28.8850
SSIM (loss 3%)	0.9549	0.9442	0.8940	0.8876
PSNR (loss 20%)	24.8095	24.7318	24.7362	24.5227
SSIM (loss 20%)	0.9065	0.8940	0.8777	0.8510

3.2.3 Effet de changement de QP :

3.2.3.1 Effet de changement de QP sur la qualité pour un PLR de 3% :

FIG.20-21 : Présentent l'impact d'un PLR de 3% pour différents QP pour les trois configurations.

On va comparer entre un résultat de QP37 et QP22 pour avoir une bonne idée, on observe que pour les trois vidéos références une chute de qualité mais d'une façon différente, pour une vidéo référence AI le PSNR diminue de 35.8495DB jusqu'à 32.9138DB donc une différence de : 2.9357DB et de 0.9759 jusqu'à 0.9063 avec une différence de 0.0696 pour le SSIM, Idem pour une vidéo référence RA on observe que le PSNR diminue et nous constatons une différence de : 1.8211DB et pour SSIM : 0.0673, enfin pour une vidéo référence LD la différence du PSNR est : 0.5814DB et 0.0605 pour le SSIM.

Ce qui nous pousse à comprendre que la configuration AI va perdre beaucoup plus de qualité que RA et LD, sauf que pour la LD elle va perdre le moins de qualité, cela relate au nombre de paquet de chaque configuration c'est-à-dire que la vidéo qui comporte le plus grand nombre de paquets va être la plus grande perdante de qualité et le contraire est vrai.

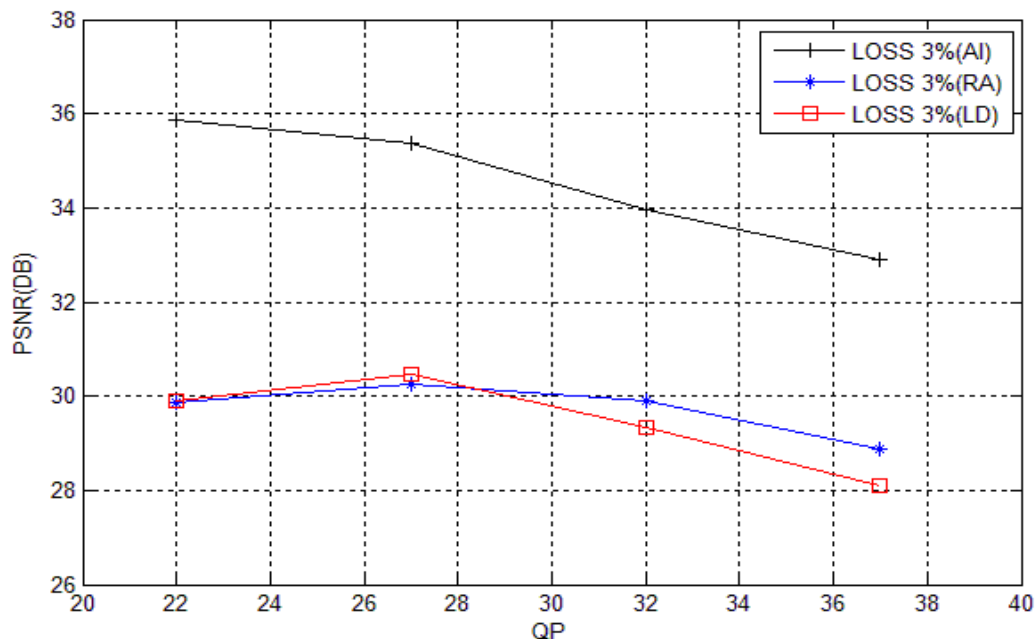


FIG.20 : (Basketball-drill) -Effet de changement de QP pour un PLR 3%

(PSNR en fonction de QP)

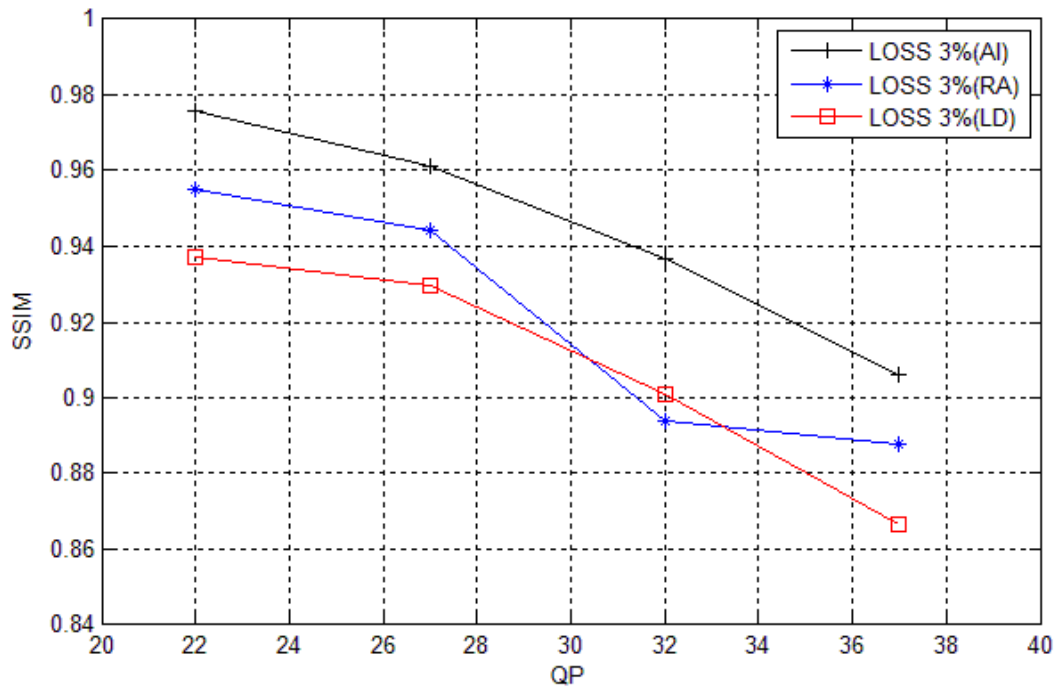


FIG.21 : (Basketball-drill) -Effet de changement de QP pour un PLR 3%

(SSIM en fonction de QP)

Tableau.9 - (BASKETBALL-DRILL-AI-RA-LD) effets de pertes de 3% :

QP	22	27	32	37
PSNR (AI)	35.8495	35.3716	33.9640	32.9138
SSIM (AI)	0.9759	0.9614	0.9367	0.9063
PSNR (RA)	29.8809	30.24306	29.8920	28.8850
SSIM (RA)	0.9549	0.9442	0.8940	0.8876
PSNR (LD)	29.9144	30.4580	29.3330	28.0933
SSIM (LD)	0.9372	0.9295	0.9008	0.8767

3.2.3.2 Effet de changement de QP sur la qualité pour un PLR de 20% :

FIG.22-23 : Présentent l'impact d'un PLR de 20% pour différents QP pour les trois configurations.

On va comparer entre un résultat de QP37 et QP22 pour avoir une bonne idée, on observe visuellement que la vidéo référence qui va perdre beaucoup plus de qualité est la LD mais on va calculer la différence pour chaque vidéo.

On remarque que pour la vidéo référence AI le PSNR va diminuer de 28.7252DB jusqu'à 27.8234DB avec une différence de 0.9018DB et pour le SSIM une diminution de 0.9549 jusqu'à 0.8792 donc une différence de 0.0757.

On va faire la même chose pour les deux autres configurations, on trouve que pour la vidéo référence RA le PSNR va diminuer de 0.2568DB et pour le SSIM de 0.0555.

Enfin pour la vidéo référence LD, on trouve une perte de PSNR est 1.966 et pour le SSIM : 0.0971.

Malgré que les trois vidéos références n'ont pas le même nombre de paquets on peut observer que pour une pertes de 20% c'est le type des paquets qui va contrôler la diminutions de la qualité, on peut remarquer que la vidéo référence LD va perdre beaucoup de qualité par rapport aux autres car son paquet est composé d'une seule image I et d'autres qui sont des P (IPPPPPP) si l'on perd un P il va affecté le P suivant jusqu'à la fin de la vidéo et on remarque que la vidéo référence AI va perdre moins de qualité car elle est composé de plusieurs I (IIIIIIII) si on perd un I il n'affecte pas le reste de la vidéo, enfin pour la vidéo référence RA une moindre perte de qualité car elle est composé de I et B mais non pas d'un seule I mais d'une succession de I et de B c'est-à-dire que le I va se répète à chaque fois (Refreshing) (IBBBIBBIBBB)

Si on perd un B ce fait affecte les suivants B jusqu'à le prochain I qui rétablie la vidéo

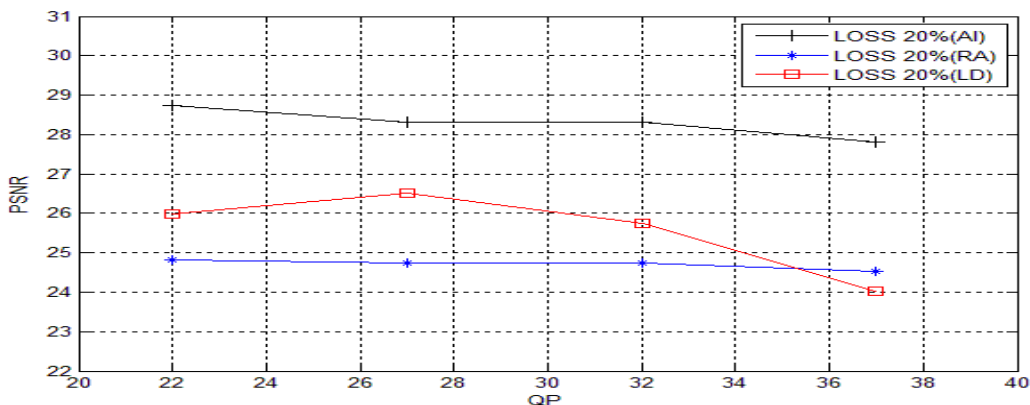


FIG.22 : (Basketball-drill) -Effet de changement de QP pour un PLR de 20%

(PSNR en fonction de QP)

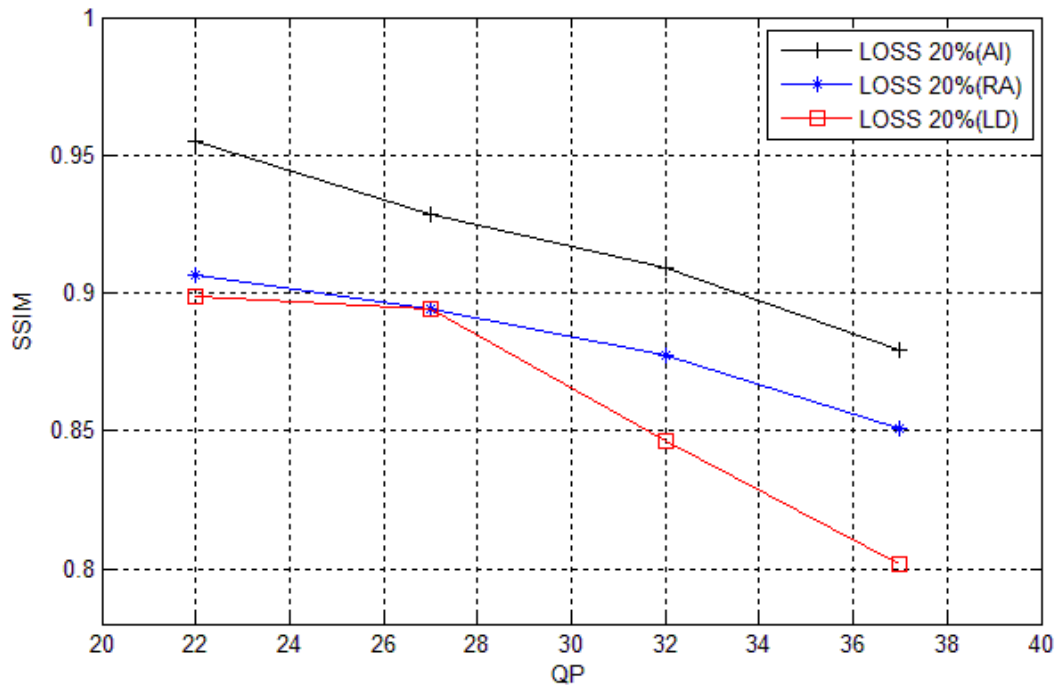


FIG.23 : (Basketball-drill) -Effet de changement de QP pour un PLR de 20%
(SSIM en fonction de QP)

Tableau.10 - (BASKETBALL-DRILL-AI-RA-LD) effets de pertes de 20% :

QP	22	27	32	37
PSNR (AI)	28.7252	28.3144	28.3131	27.8234
SSIM (AI)	0.9549	0.9282	0.9089	0.8792
PSNR (RA)	24.8095	24.7318	24.7362	24.5227
SSIM (RA)	0.9065	0.8940	0.8777	0.8510
PSNR (LD)	25.9860	26.5013	25.7329	24.0200
SSIM (LD)	0.8989	0.8944	0.8462	0.8018

3.2.4 Etude de l'impact de changement du QP sur la Qualité pour les trois configurations pour différents PLR :

FIG.24-25-26-27-28-29 : représentent l'impact de changement de PLR pour les quatre QP, on remarque qu'à chaque fois que nous augmentons le pourcentage de la perte, la qualité diminue pour les trois configurations, on peut voir aussi d'après les figures FIG.24-26-28 pour les deux pertes le SSIM de QP 22 est toujours grand par rapport aux autres QP, et le SSIM de QP 27 est toujours grand par rapport aux QP 32 et ce dernier est toujours plus grand par rapport au QP 37 tout ça grâce au nombre de paquets pour chaque QP.

On ne peut pas voir ces diminutions au figures FIG.23-25-27 car j'ai utilisé pour cette étude (après transmission) un type de perte BURST qui ne permet pas de donner des résultats précis avec le calculateur PSNR par contre en utilisant le SSIM j'ai pu faire une étude.

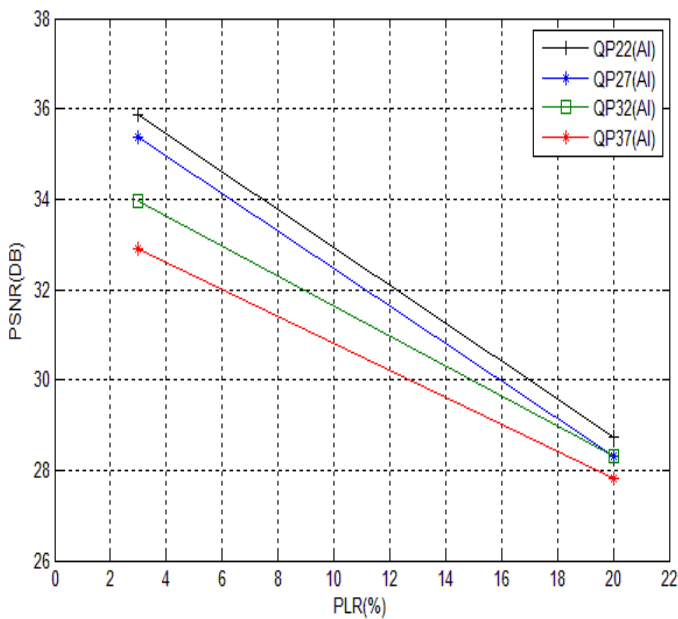


FIG.24 : (Basketball-drill-AI)

-Impact de PLR pour les quatre QP

(PSNR en fonction de PLR)

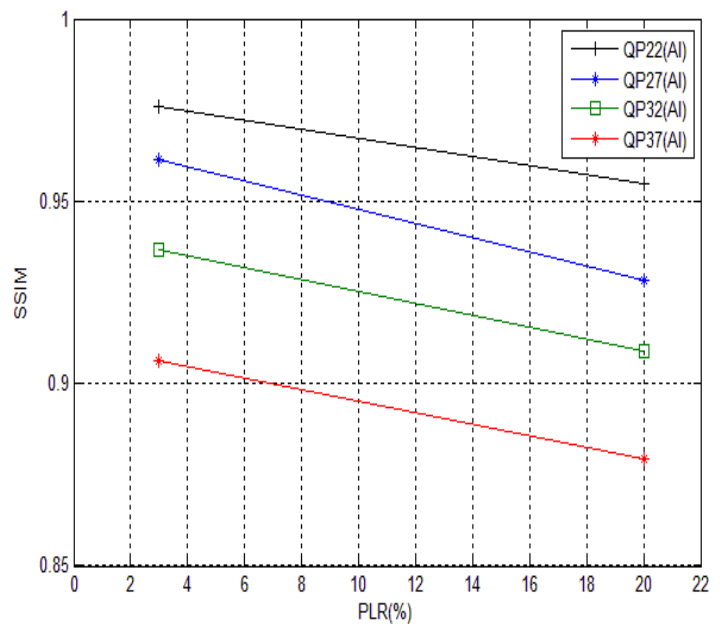


FIG.25 : (Basketball-drill-AI)

-Impact de PLR pour les quatre QP

(SSIM en fonction de PLR)

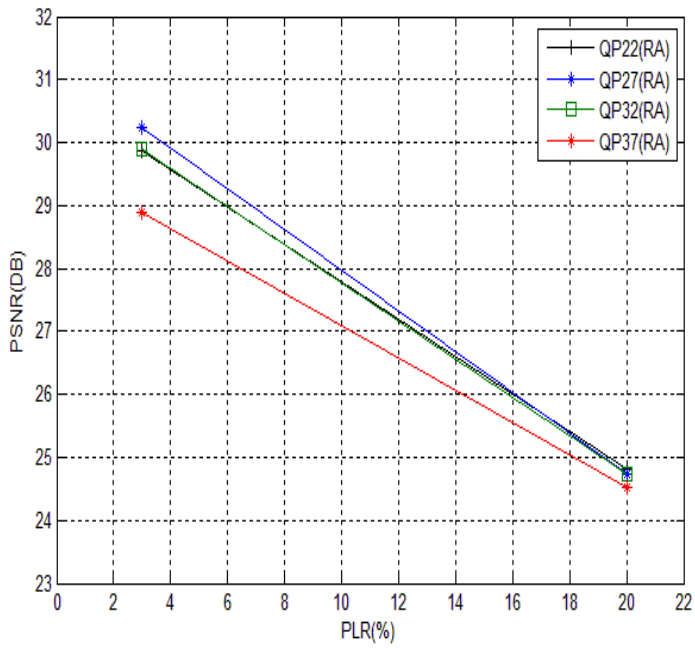


FIG.26 : (Basketball-drill-RA)
-Impact de PLR pour les quatre QP
(PSNR en fonction de PLR)

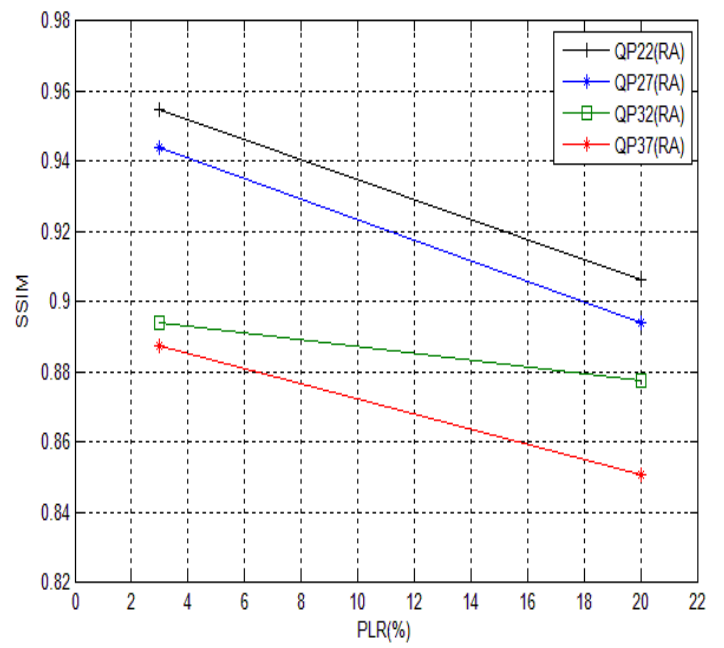


FIG.27 : (Basketball-drill-RA)
-Impact de PLR pour les quatre QP
(SSIM en fonction de PLR)

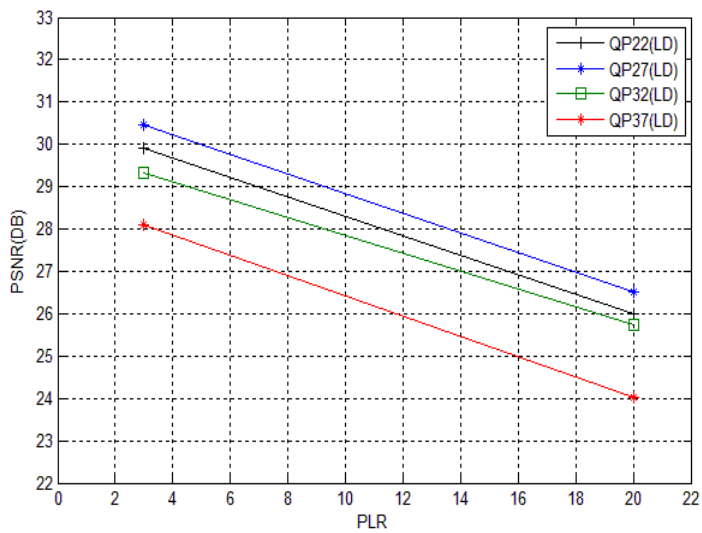


FIG.28 : (Basketball-drill-LD)
-Impact de PLR pour les quatre QP
(PSNR en fonction de PLR)

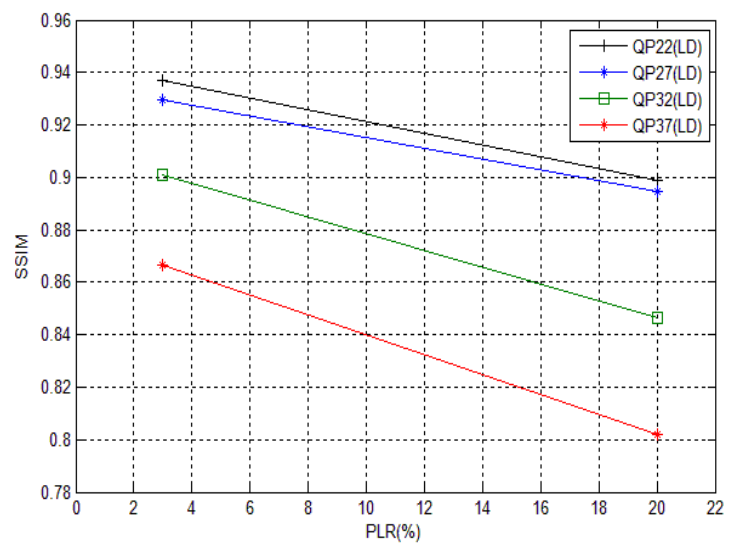


FIG.29 : (Basketball-drill-LD)
-Impact de PLR pour les quatre QP
(SSIM en fonction de PLR)

Tableau.11 - (BASKETBALL-DRILL-AI-RA-LD) : l'impact de changement du QP sur la Qualité pour les deux PLR (3 et 20%) :

PLR	3% (QP22)	20% (QP22)	3% (QP27)	20% (QP27)	3% (QP32)	20% (QP32)	3% (QP37)	20% (QP37)
PSNR(AI)	35.8495	28.7252	35.3716	28.3144	33.9640	28.3131	32.9138	27.8234
SSIM(AI)	0.9759	0.9549	0.9614	0.9282	0.9367	0.9089	0.9063	0.8792
PSNR (RA)	29.8809	24.8095	30.24306	24.7318	29.8920	24.7362	28.8850	24.5227
SSIM (RA)	0.9549	0.9065	0.9442	0.8940	0.8940	0.8777	0.8876	0.8510
PSNR (LD)	29.9144	25.9860	30.4580	26.5013	29.3330	25.7329	28.0933	24.0200
SSIM (LD)	0.9372	0.8989	0.9295	0.8944	0.9008	0.8462	0.8767	0.8018

3.3 Conclusion :

Dans ce chapitre nous avons fait une étude sur l'impact de changement de facteur QP sur la qualité vidéo avant et après la transmission. Nous avons conclu que l'augmentation du QP entraîne une perte des paquets, ce qui va diminuer la qualité de la vidéo. Nous avons remarqué aussi que les pertes affectent les trois configurations d'une manière différente où le plus touché est la vidéo référence LD. Ainsi nous avons constaté à travers les résultats obtenus que le SSIM est le meilleur calculateur par rapport au PSNR.

Conclusion générale :

Au cœur de ce projet de fin d'étude, nous avons organisé notre étude en trois parties. Nous avons d'abord étudié la compression des données (images, vidéos), puis nous avons expliqué les types de compression qui existent, et les normes (H.264, H.265) et ses utilisations. Nous avons aussi expliqué les deux métriques PSNR et SSIM.

Nous avons présenté JM, le logiciel référence de la norme H.264/AVC développé par Karsten.Suehring. La dernière version que nous avons utilisée est le JM 19. Ce logiciel peut être utilisé sur Windows ou Linux. Avec ce logiciel on peut obtenir une vidéo référence avec un taux de compression choisi.

Ensuite nous avons expliqué EvalVid, un simulateur qui contient plusieurs outils permettant d'évaluer la qualité de la vidéo transmise sur un réseau simulé.

Plusieurs perspectives peuvent être envisagées sur la base de ce travail. Nous les structurons comme suit :

- Continuer à faire des études sur plusieurs vidéos avec les trois configurations (AI-RA-LD).
- Intégrer EvalVid avec NS (Network Simulator) pour avoir un autre type de transmission ce qui va donner autres résultats.
- Utiliser les PLR (3%, 5%, 10%, 20%) pour toutes les vidéos pour avoir une idée précise.

Références bibliographiques :

[1] L'encyclopédie libre Wikipédia

https://fr.wikipedia.org/wiki/Compression_de_donn%C3%A9es ,4 mai 2019.

[2] L'encyclopédie libre Wikipédia

https://en.wikipedia.org/wiki/Image_compression, 18 mars 2019.

[3] Slimane Larabi, “Cours : SYSTEMES MULTIMEDIA, Chapitre 4 », Master RSD, 2014/2015.

[4] Selmi Mohammed El-Amin et Yala Mohammed. “Synthèse et étude comparative sur les méthodes de compression vidéo » Faculté des Sciences département d'Informatique, Abou Bakr Belkaid -Tlemcen, 2017.

[5] TOUNSI Billal et ZIDOUNE Halim “Compression de données”, Faculté des Sciences Exactes département d'Informatique, Université A/Mira de Bejaïa 2016.

[6] Marco Cagnazzo “Les standards H.264/AVC et HEVC”, Département Traitement du Signal et des Images, TELECOM ParisTech, 24 mai 2013

[7] <https://www.vcodex.com/an-overview-of-h264-advanced-video-coding/>.

[8] <https://www.vcodex.com/h265-high-efficiency-video-coding-hevc/>.

[9] Mohd.A.Saleh “Review for High Efficiency Video Coding (HEVC)”, December 2014

[10] <http://iphome.hhi.de/suehring/>.

[11] Alexis Michael Tourapis,” H.264/14496-10 AVC Reference Software Manual” Jan, 2009.

[12] L'encyclopédie libre Wikipédia https://en.wikipedia.org/wiki/Peak_signal-to-noise_ratio, 14 Mars 2019

[13] Gabriel Prieto Renieblas et Agustín Turrero Nogués et Alberto Muñoz González et Nieves Gómez-Leon Eduardo Guibelalde del Castillo “Structural similarity index family for image quality assessment in radiological images”, Sep 2017.

[14] AHMED SEGHIR Zianou , HACHOUF Fella , ARDJOUNE Mourad , FALEK Salem “Etude comparative des méthodes d'évaluation de la qualité d'image” Institut des sciences et technologies, Centre Universitaire Abbas Laghrour Khenchela.

[15] Jirka Klaue, “EvalVid - A Video Quality Evaluation Tool-set” , 09.08.11.

[16] Chih-Heng, Ke, “How to evaluate H.263/H.264/MPEG4 video transmission using the NS2 simulator?” Phd candidate, EE Department, NCKU, Taiwan, 2005/12/22.

[17] EvalVid with GPAC – Usage

<http://www2.tkn.tuberlin.de/research/evalvid/EvalVid/docevalvid.html>.