

Décodeur Video HEVC basse consommation

Erwan NOGUES, Erwan RAFFIN, Maxime PELCAT, Daniel MÉNARD

Institut d'Électronique et de Télécommunications de Rennes - UMR CNRS 6164
20 Avenue des Buttes de Coesmes, 35708 Rennes, France

erwan.nogues@insa-rennes.fr, erwan.raffin@insa-rennes.fr
maxime.pelcat@insa-rennes.fr, daniel.menard@insa-rennes.fr

Résumé – Avec toujours plus de fonctionnalités à supporter, l'autonomie des systèmes embarqués est aujourd'hui sous pression. Parmi ces fonctionnalités, le nouveau standard de compression HEVC permet d'augmenter le niveau de compression de 50% en moyenne tout en conservant le même niveau de qualité subjective que son prédécesseur, H.264. Dans cette étude, nous proposons une implémentation modifiée du décodeur HEVC permettant de réduire sa consommation d'énergie jusqu'à 28% par rapport à une implémentation logicielle standard.

Abstract – With ever more features to support, the autonomy of embedded systems is under a great pressure. Amongst the new supported features, HEVC video compression standard is to replace H.264 as the dominating standard since HEVC doubles video compression rates for the same subjective image quality, widening the scope of video applications. In this paper, a modified HEVC decoder is proposed to reduce the energy consumption with a limited impact on the distortion. A software implementation of the modified decoder can save up to 28% of energy.

1 Introduction

1.1 Contexte

Aujourd'hui, les appareils électroniques offrent de plus en plus de fonctionnalités (vidéo, audio, GPS, internet) et des connectivités variées (multi-systèmes de radio avec WiFi, Bluetooth, UMTS, HSPA, LTE-advanced ...). La demande en puissance de ces appareils est donc grandissante pour la partie numérique et notamment le processeur. Pour répondre à ce besoin sans cesse croissant de nouvelles fonctionnalités et donc de puissance de calcul, les architectures des processeurs ont beaucoup évolué : processeurs multi-coeurs, GPU et autres accélérateurs matériels dédiés. Cependant, alors que de nouvelles architectures matérielles peinent à répondre aux exigences de performance, l'évolution de la technologie des batteries est quant à elle encore plus lente. Ainsi, l'évolution des batteries n'a pas encore permis une rupture technologique suffisante pour pallier la demande actuelle en énergie. En conséquence, l'autonomie des systèmes embarqués est aujourd'hui sous pression. Parmi les nouveaux services supportés par les terminaux mobiles, la vidéo prend une place prépondérante. En effet, des analyses récentes de tendance montrent qu'elle représentera 70% du trafic internet mobile dès 2016. Accompagnant cette croissance, de nouvelles technologies émergent permettant de nouveaux services et applications. Parmi elles, High Efficiency Video Coding (HEVC) permet de doubler la compression de données tout en garantissant une qualité subjective équivalente à son prédécesseur, la norme H.264. Cela permet d'élargir le spectre des applications possibles. De plus, le groupe de standardisation MPEG au sein de l'ISO a lancé des travaux visant à favoriser des implantations efficaces en consommation d'éner-

gie de décodeurs vidéo [1]. De ce cadre, le décodeur peut notamment adapter la qualité de la vidéo décodée en fonction de l'énergie nécessaire.

Dans cette étude, un décodeur HEVC modifié est proposé incluant des modes de complexité réduite visant à améliorer l'efficacité énergétique du terminal. Puisque la séquence d'entrée n'est pas modifiée, chaque terminal peut adapter son mode de décodage en fonction de ses capacités énergétiques individuelles. La suite du papier est organisée comme suit : après avoir présenté l'état de l'art dans la section 2, la section 3 décrit le décodeur HEVC modifié. Dans la section 4, les résultats expérimentaux sont donnés sur une plate forme classiquement utilisée pour des terminaux mobiles. Enfin, les conclusions sont présentées dans la section 5.

2 Etat de l'art

Récemment, un certain nombre d'études a été publié sur la réduction d'énergie liée au décodage vidéo que ce soit au niveau architectural de la définition des systèmes ou au niveau des applications elles-mêmes.

Tout d'abord, plusieurs méthodes [4, 5, 8] ont été proposées afin d'utiliser la capacité des plate formes matérielles sur lesquelles reposent l'application décodage vidéo. Une première classe d'études cherche à utiliser la capacité des processeurs à pouvoir adapter leur fréquence de fonctionnement et leur tension d'alimentation aux besoins réels de l'application. Appelé "Dynamic voltage and frequency scaling (DVFS)", ce mécanisme permet de réduire la puissance dynamique des circuits et se prête particulièrement bien au décodage vidéo. En effet, dans le cas de décodage vidéo en temps réel, la vitesse d'affichage

est plus faible que la vitesse de décodage potentielle, générant un temps de repos entre le décodage et l’affichage. Ce temps de repos est utilisé pour faire des économies d’énergie. Conjointement à ces techniques, la parallélisation des traitements de décodage permet de dégager encore plus de temps de repos.

Sous un autre angle d’optimisation, d’autres méthodes consistent à adapter le couple consommation/distorsion en fonction du niveau d’exigence de l’utilisateur et aux ressources disponibles [6, 9]. Par exemple, il est proposé soit d’utiliser une résolution plus faible permettant ainsi de faire des gains en consommation sur le décodage, soit de désactiver des parties de décodeur temporairement.

Dans ce papier un nouveau mécanisme de décodage vidéo est étudié. Il vise à améliorer l’efficacité énergétique des terminaux mobiles en réduisant la complexité du décodage.

3 Description du décodeur HEVC

3.1 Décodage HEVC

Dans ce papier, nous utilisons une structure standard du décodeur HEVC [10]. Celui-ci se divise en plusieurs parties comme décrites dans la figure 1. Dans un premier temps, le décodage entropique permet l’extraction des éléments de syntaxe de la séquence d’entrée. Ensuite, les données sont dé-quantifiées puis transformées par une Transformée en Cosinus Discrète (TCD) inverse. La prédiction est alors appliquée et peut être de deux types : soit *intra*, soit *inter* en fonction du type d’encodage de la séquence d’entrée. Dans la cas de la prédiction *inter*, elle est calculée en fonction des autres trames reçues et le décodage se fait grâce aux vecteurs de mouvement. Enfin les filtres de boucle sont appliqués pour réduire les éventuels artefacts et améliorer la qualité de l’image.

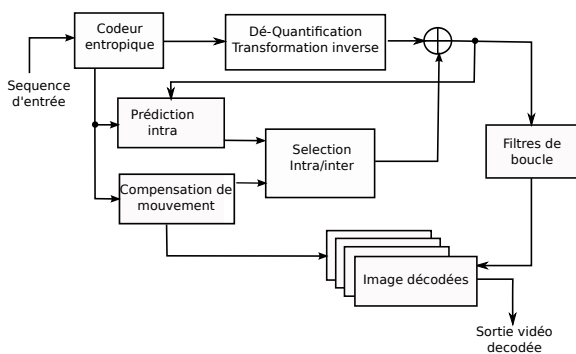


FIGURE 1 – Version simplifiée du décodeur HEVC standard

La réduction d’énergie peut se faire de plusieurs façons notamment si l’on autorise une légère distorsion de qualité. Lorsqu’on analyse les répartitions de charge du décodeur HEVC on peut remarquer que les procédures de filtrage prennent une place prépondérante dans le total quelque soit le type de processeur utilisé [2, 3]. Dans cette étude, nous proposons de mo-

difier les caractéristiques des filtres utilisés afin de réduire la complexité de décodage et donc d’améliorer l’efficacité du décodage vidéo.

3.2 Génération de filtres simplifiés

La proposition est donc réduire la complexité des filtres comme décrit dans la figure 2. Cette réduction passe par la reformulation des coefficients de filtres. Dans la suite de l’étude, l’exemple des filtres d’interpolation est décrit.

Pour faire l’estimation de mouvements sur des fractions de pixel, des filtres d’interpolation 1-D sont utilisés dans HEVC. L’estimation de mouvement est utilisée par l’encodeur vidéo pour compresser la séquence à transmettre en exploitant la redondance temporelle entre les différentes trames de celle-ci. Un vecteur de mouvement est défini comme la position relative entre la trame à prédire et la trame de référence. Cependant, le mouvement des objets étant continu il ne peut correspondre exactement à des positions entières de pixel. C’est pourquoi en utilisant une estimation de mouvement fractionnaire (au sous pixel) l’estimation de vecteurs de mouvement est rendue plus précise et réduit l’erreur résiduelle. Kemal et al. analysent en détail la génération des filtres d’interpolation en fractions de pixel : soit $\{p_i\}(i = M_{min}, \dots, M_{max})$ of $Taille = M_{max} - M_{min} + 1$, la TCD permet d’obtenir le coefficient de Fourier C_k (Eq. 1). La paire transformation et transformation inverse peut être pré-calculée et fusionnée pour générer les positions fractionnaires [7].

$$C_k = \frac{2}{Taille} \cdot \sum_{l=M_{min}}^{M_{max}} p(l) \cos\left(\frac{(2 \cdot l - 2 + Taille) \cdot k \cdot \pi}{2 \cdot Taille}\right) \quad (1)$$

Le standard HEVC utilise des filtres filtre à Réponse Impulsionnelle Finie (RIF) pour faire l’interpolation des pixels. Tseng et al. [11] proposent une méthode de synthèse de ces filtres à partir des coefficients de Fourier de l’équation 1. Ces filtres peuvent posséder un nombre pair ou impair de coefficients en fonction du type d’échantillons à interpoler : luminance ou chrominance. L’équation 2 donne l’exemple de génération de filtres pairs.

$$filtre_m(\alpha) = \frac{1}{M} \cos\left(\pi \frac{m - \alpha}{N - 1}\right) \sum_{k=0}^{2M-1} (c_k^2 \cdot \phi_k) \quad (2)$$

$$\text{with } \phi_k = \cos\left(\frac{(2m-1+2M)\pi k}{4M}\right) \cos\left(\frac{(2\alpha-1+2M)\pi k}{4M}\right).$$

Alors que le décodeur standardisé propose une configuration fixe de filtre, la solution proposée dans ce papier consiste à générer des filtres avec une taille variant de un coefficient jusqu’au nombre de coefficients du standard HEVC. Actuellement, les filtres liés à la luminance utilisent $M_{min} = -3$, $M_{max} = 4$ et $Taille = 8$ alors que les filtres de la chrominance utilisent $Taille = 4$.

Dans cette étude, quatre classes sont définies relativement à leur complexité : faible, moyenne, intermédiaire et forte avec

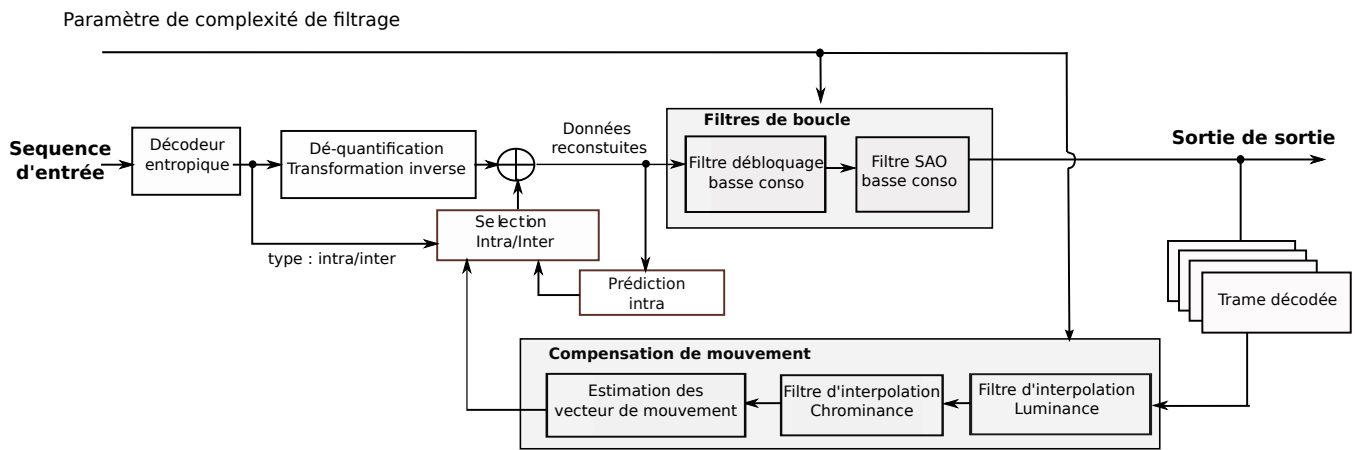


FIGURE 2 – Décodeur HEVC modifié avec filtres basse consommation

respectivement $Taille = 1, 3, 5, 8$ pour la luminance et avec $Taille = 1, 2, 3, 4$ pour la chrominance. Les tableaux 1 et 2 décrivent les filtres utilisés pour réduire la complexité.

TABLE 1 – Filtre d'interpolation pour la luminance : version standard et modifiée avec une taille variable

	$\alpha = 1/4$	$\alpha = 1/2$
Standard	-1, 4, -10, 58, 17, -5, 1	-1, 4, -11, 40, 40, -11, 4, -1
Taille=7	-1, 4, -10, 58, 17, -5, 1	-1,4, -11, 40, 40, -11, 3
Taille=5	1, -6, 20, 54, -5	2, -9, 40, 40, -9
Taille=3	-4, 20, 48	-9, 41, 32
Taille=1	64	64

TABLE 2 – Filtre d'interpolation pour la chrominance : version standard et modifiée avec une taille variable

α	Standard	Taille=1	Taille=2	Taille=3
1/8	-2, 58, 10, -2	64	58, 7	-3, 62, 5
1/4	-4, 54, 16, -2	64	50, 15	-5, 58, 11
3/8	-6, 46, 28, -4	64	41, 23	-7, 51, 20
1/2	-4, 36, 36, -4	64	32, 32	-6, 42, 28

Afin de faciliter les comparaisons de performance, les différents filtres sont rassemblés en catégories et décrits dans le tableau 3.

TABLE 3 – Taille des filtres associée aux configurations

Configuration	Chroma filter size	luma filter size
faible	1	1
moyenne	2	3
intermédiaire	3	5
forte	3	7

4 Résultats expérimentaux

4.1 Banc de mesures

Les performances du système sont évaluées sur une plateforme Exynos 5410. Celle-ci est communément utilisée par les terminaux mobiles car elle est basée sur la technologie ARM big.LITTLE développée pour les applications mobiles basse consommation à base de processeurs généralistes. Cette plateforme adapte notamment sa fréquence de fonctionnement à la charge des processeurs réduisant ainsi l'énergie consommée. Les mesures de consommation sont réalisées par des capteurs placés sur le SoC qui permettent de collecter la puissance des processeurs et la mémoire avec une fréquence de 10 Hz.

4.2 Séquences de test

Pour obtenir une vue complète de la solution proposée, une séquence de test dédiée est proposée permettant d'étudier l'impact de différentes scènes sur le codage. Elle concatène les séquences standardisées pour HEVC : *BasketBall Drive*, *Cactus* et *BQ Terrace* comme décrit dans la figure 3.

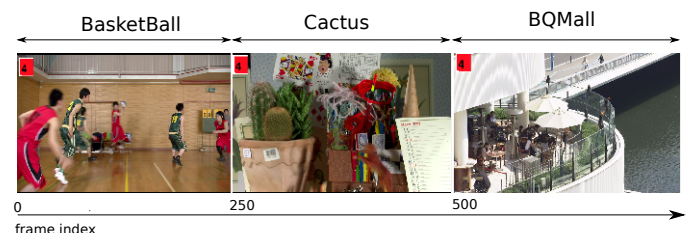


FIGURE 3 – Séquence de test

La résolution choisie est celle de la haute définition 1920x1080 avec un débit de 3.3 Mb/s. L'image servant de référence à la prédiction est envoyée toutes les secondes à un rythme de 25 trames par seconde.

4.3 Résultats

Dans cette section, la solution proposée est évaluée sous deux critères. Tout d'abord, les gains en énergie sont présentés. Puis, la distorsion par rapport à la solution standard est donnée. La table 4 résume l'énergie nécessaire pour décoder la séquence de test en fonction du mode de décodage. On peut alors mesurer les gains en énergie réalisés par la solution proposée par rapport à la solution standard en fonction du mode de filtrage.

TABLE 4 – Gains en énergie

Configuration	Energie (J)	Gains (%)
Standard	88.02	-
basse	63.02	28.37
moyenne	69.89	20.57
intermédiaire	72.37	17.79
grande	80.96	7.99

Les mesures montrent que des gains allant jusqu'à 28 % peuvent être atteints sur l'énergie nécessaire pour décoder la séquence. On remarque que les gains sont aussi directement liés à la complexité de filtrage choisie.

Dans un second temps, la qualité de décodage est estimée en calculant le Peak Signal-to-Noise Ratio (PSNR) trame par trame entre l'image décodée par le schéma standard et la méthode proposée.

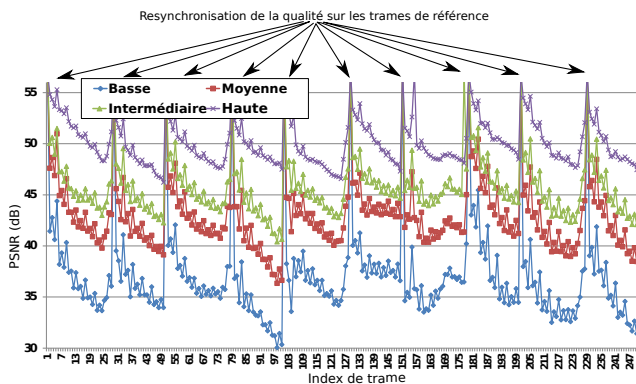


FIGURE 4 – PSNR par trame sur la séquence Basket Ball drive

La figure 4 trace le PSNR en fonction de l'index de trame décodée. On peut remarquer que la qualité est naturellement plus dégradée avec les filtres les moins complexes. Par ailleurs, on note aussi une dérive de la dégradation avec le temps et que cette dérive est stoppée à l'apparition d'une image de référence, toutes les 25 trames dans l'exemple. Cette caractéristique est intéressante pour la conception du système car elle permet au décodeur de maîtriser la dérive de qualité par ce mécanisme.

5 Conclusion

Dans ce papier, une variante du décodeur HEVC est présentée améliorant son efficacité énergétique. La méthode proposée

permet de gagner jusqu'à 28% d'énergie par rapport à un décodeur classique sur une plateforme communément utilisée pour les terminaux mobiles. Ces gains sont obtenus au prix d'un compromis sur la qualité. Cependant ce compromis peut-être contrôlé par l'insertion régulière de trames de référence.

Références

- [1] Context, objectives, use cases and requirements for green mpeg. *SO/IEC JTC1/SC29/WG11/N13468*, April 2013.
- [2] F. Bossen, B. Bross, K. Suhling, and D. Flynn. Hevc complexity and implementation analysis. *Circuits and Systems for Video Technology, IEEE Transactions on*, 22(12):1685–1696, 2012.
- [3] M. Chavarrias, F. Pescador, M. Garrido, M. Raulet, et al. A dsp-based hevc decoder implementation using an actor language dataflow model. *Consumer Electronics, IEEE Transactions on*, 59(4):839–847, 2013.
- [4] J. Cho, I. Cho, D.-Y. Kim, and B. Lee. A combined approach for qos-guaranteed and low-power video decoding. *Consumer Electronics, IEEE Transactions on*, 57(2):651–657, 2011.
- [5] W. Hamidouche, M. Raulet, and O. Deforges. Parallel shvc decoder : Implementation and analysis. *IEEE conference on ICME*, 2014.
- [6] Y. He, M. Kunstner, S. Gudumasu, E.-S. Ryu, Y. Ye, and X. Xiu. Power aware hevc streaming for mobile. In *Visual Communications and Image Processing (VCIP), 2013*, pages 1–5. IEEE, 2013.
- [7] U. Kemal, A. Alshin, E. Alshina, F. Bossen, W. Han, J. Park, and J. Lainema. Motion compensated prediction and interpolation filter design in h. 265/hevc. 2013.
- [8] N. Mastronarde, K. Kanoun, D. Atienza, P. Frossard, and M. van der Schaar. Markov decision process based energy-efficient on-line scheduling for slice-parallel video decoders on multicore systems. *Multimedia, IEEE Transactions on*, 15(2):268–278, 2013.
- [9] E. Nogues, S. Holmbacka, M. Pelcat, D. Menard, and J. Lilius. Power-aware hevc decoding with tunable image quality. In *Signal Processing Systems (SiPS), 2014 IEEE Workshop on*, pages 1–6. IEEE, 2014.
- [10] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand. Overview of the high efficiency video coding (hevc) standard. *Circuits and Systems for Video Technology, IEEE Transactions on*, 22(12):1649–1668, 2012.
- [11] C.-C. Tseng and S.-L. Lee. Design of fractional delay fir filter using discrete cosine transform. In *Circuits and Systems, 2008. APCCAS 2008. IEEE Asia Pacific Conference on*, pages 858–861. IEEE, 2008.