International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-10 Issue-5, January 2022

# DSP TMS320C6678 Based SHVC Encoder Implementation and its Optimization

**Ibtissem Wali, Amina Kessentini, Mohamed Ali Ben Ayed, Nouri Masmoudi**

***Abstract:*** ***The programmable processors newest technologies, as for example the multicore Digital Signal Processors (DSP), offer a promising solution for overcoming the complexity of the real time video encoding application. In this paper, the SHVC video encoder was effectively implemented just on a single core among the eight cores of TMS320C6678 DSP for a Common Intermediate Format (CIF)input video sequence resolution(352x288). Performance optimization of the SHVC encoder had reached up 41% compared to its reference software enabling a real-time implementation of the SHVC encoder for CIF input videos sequence resolution. The proposed SHVC implementation was carried out on different quantization parameters (QP). Several experimental tests had proved our performance achievement for real-time encoding on TMS320C6678.***

***Keywords: Scalable Video Coding, SHVC encoder, DSP-TMS320C6678 algorithm optimization.***

## I. INTRODUCTION

Nowadays, different types of telecommunication and multimedia networks are steadily speeding-up. In this context, video coding, one of the most demanding tasks in terms of computational load and energy consumption, permits a tradeoff between energy consumption and video quality. Digital video transmission over the internet has also been increasingly used in numerous applications. Hence, the SHVC standard [1], defined as the scalable extension of the high efficiency video coding standard, is held for a wide range of applications. Moreover, SHVC achieves a high video quality with an efficient data compression when compared to other existing new video coding standards[2]. Coupled with the existing HEVC features, new video coding tools are added to SHVC, making it a more efficient standard. However, this efficiency entails an additional increase in complexity for both encoding and decoding processes [2].

Manuscript received on March 02, 2020.
Revised Manuscript received on November 24, 2021.
Manuscript published on January 30, 2022.
* Correspondence Author
**Ibtissem Wali*,** University of Sfax, National Engineering School of Sfax Electronics and Information Technology Laboratory, LETI Sfax, Tunisia E-mail. wali.ibtissem@gmail.com
**Amina Kessentini,** University of Sfax, National Engineering School of Sfax Electronics and Information Technology Laboratory, LETI Sfax, Tunisia
**Mohamed Ali Ben Ayed,** University of Sfax, National Engineering School of Sfax Electronics and Information Technology Laboratory, LETI Sfax, Tunisia
**Nouri Masmoudi,** University of Sfax, National Engineering School of Sfax Electronics and Information Technology Laboratory, LETI Sfax, Tunisia



Due to the computational complexity of this standard, several implementation solutions are performed on different processors and platforms [2]. However, this implementation still may not satisfy real-time constraints that can be overcome using a high clock frequency and large size memory platforms. Fortunately, the Digital Signal Processors (DSP) latest generation may support the video coding encoder at a relatively low cost. Indeed, DSPs are specialized microprocessors designed to perform embedded digital signal processing algorithms and, thus, massively decrease power consumption, size, and price [3]. Furthermore, these microprocessors are programmable; thus, they are considerably flexible. However, their complex architecture causes immense difficulties. Obviously, this cost-efficiency feature makes DSP programming not only an important but also a challenging issue for discussion. This paper, which presents a DSP implementation of a real-time SHVC encoder, is divided into V sections organized as follows. The SHVC encoder is presented in Section II. The TMS320C6678 architecture is detailed in Section III. In Section IV, SHVC software optimization is performed and the decoder performance is outlined. Results of the profiling tests are discussed in Section V. Finally, a conclusion and perspectives are presented.

## II. PREVIOUS WORK

The latest video coding standard has clearly achieved a satisfactory coding efficiency. However, the computational complexity cost has posed a critical issue, especially in the implementation phase. Therefore, a number of efforts have recently been made to explore new generation platforms in order to reduce complexity and speed up video coding. Several implementations were performed using H.264/SVC, HEVC and SHVC.

### 1. Previous standard implementation:

Authors in [4]suggested a DSP-based video decoder implementation compliant to H.264/SVC standard, improving the initial version performance about 40% within performance optimizations and reaching real-time for CIF sequences. Moreover, in[5], a multi-DSP based video decoder was accomplished in accordance with the H.264/SVC standard. Two optimization processes were presented: the frame decoding process and the deblocking filter one. An average improvement of 14.8% was achieved. Two different Texas Instruments platforms were used in [6] in order to embed the HEVC encoder on both ARM processor and TMS30C6678 DSP. Results showed a saving up to 62% of the encoding time.

In [7], HEVC encoder software implementation and an optimized architecture were presented on a DSP TMS320C6678 single core. The authors proposed using single instruction multiple data (SIMD) operations and data level parallelism in order to optimize both Sum of Absolute Differences (SAD) and Sum Square Error (SSE) engines. Results showed approximately a gain of 24% compared to HEVC reference model.

## 2. SHVC implementation

SHVC implementation was performed in[8] where authors used optimized SIMD operations methods for the x86 architecture using the SSE4 technology. Parallelism tools, including frame based parallelism and wave front parallelism, were applied on the SHVC decoder implementation. This hybrid solution led to a high speed up performance with a good tradeoff between frame latency and decoder memory usage.As shown in [9], SHVC standard was encapsulated with the Project on Advanced Content multimedia library (GPAC), which is an open-source software dedicated to rich-media and broadcast technologies, into an mp4 MPEG-2 content, transferred by the sender side. At the receiver side, the GPAC got the SHVC packets of video by SHVC decoder. The GPAC allows a switch between displaying the BL and the EL through an interactive interface. Based on the specificities of SHVC, which is a multi-loop coding structure requiring a total decoding of intermediate layers, authors in [10] were led to propose a parallel processing architecture to enhance the decoding latency and frame rate. Results lead to an acceptable performance on speed up with a good balance between decoding time, latency and memory usage. When analyzing these previous works, we notice that multiple implementations for H.264/SVC and beyond were embedded on several platforms. However, almost all the proposed SHVC works addressed the decoding side since the encoder implementation can be challenging due to its computational complexity. Consequently, a new contribution, which focuses on the SHVC encoder implementation and its optimization, is emphasized in this paper.

## III. DSP C6678 ARCHITECTURE

In this study, a DSP-based solution was opted for implementation purpose mainly because of its programming flexibility, its comparatively low cost as well as its low consumption. The latest generation of the TMS320C6678 Digital Signal Processor, which uses the fixed-point precision and the floating-point one, developed by Texas Instruments (TI) [14]is actually targeted in this work. Each DSP processor uses up to 1.25 GHz. Thanks to this high performance, they can be effectively used for multimedia applications including SHVC. Moreover, owing to their developed set of tools, time design and conception for this kind of applications is reduced. The TMS320C6678 architecture is characterized by eight embedded cores.

Each core is based on the Very Long Instruction Word (VLIW) architecture. VLIW architecture takes advantage of numerous levels of parallelism. The instruction level parallelism comes from the eight functional units arranged in two sides as shown in figure1. The functional diagram of the C66x DSP core architecture is depicted in figure1. In fact, each side has four units: L, M, S and D. Multiplication operation is performed on M unit. However, the D unit focuses on loading/storing and address manipulation. The L and S units perform addition, subtraction, logical, and branching operations. This architecture may, hence, issues eight parallel instructions for every cycle. In addition, there are two sets of 32-bit registers for a total of 64 registers.

Furthermore, a thread-level parallelism may be exploited by running different threads. Each core integrates a large amount of memory structure. There is a two-level memory system: 32 KB of L1 data and program cache with 512 KB of L2 unified cache. Two L1 and L2 on-chip memories could be dynamically configured as cache, RAM or a combination of the two. L2 shared memory sized 4096 KB, used as a shared memory between the eight cores, is defined in addition to L1 and L2 cache memories. TMS320C6678 defines a 64-bits DDR3 external memory which operates up to 1600 MHz. EDMA (Enhanced Direct Memory Access) is used for transferring and exchanging data[14].

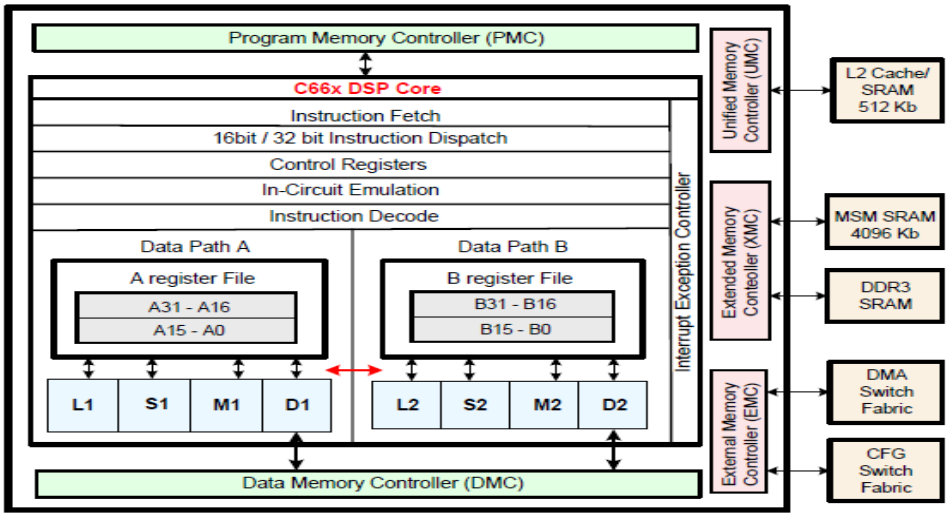



**Fig.1 Functional diagram of the C66x DSP core architecture**



Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

## IV. SHVC AND OPTIMIZED SHVC

### 1. SHVC

Scalable video coding offers a mechanism for multilayer video coding. Each layer is responsible for supporting a certain quality in the same video. This mechanism integrates one Base Layer (BL), which is in charge of carrying the lowest quality and up to seven Enhancement Layers ELs. They could be coded by referencing lower layers to improve the video quality. Moreover, the term "scalability" is based on removing parts of the video bit stream in rank to adjust it to the various end user needs. Consequently, a video bit stream is considered scalable while parts of the stream may be unconcerned in a way that the resulting sub-stream forms a further valid bit stream for another target decoder. A two-layer scalability scheme is presented in Fig 2.

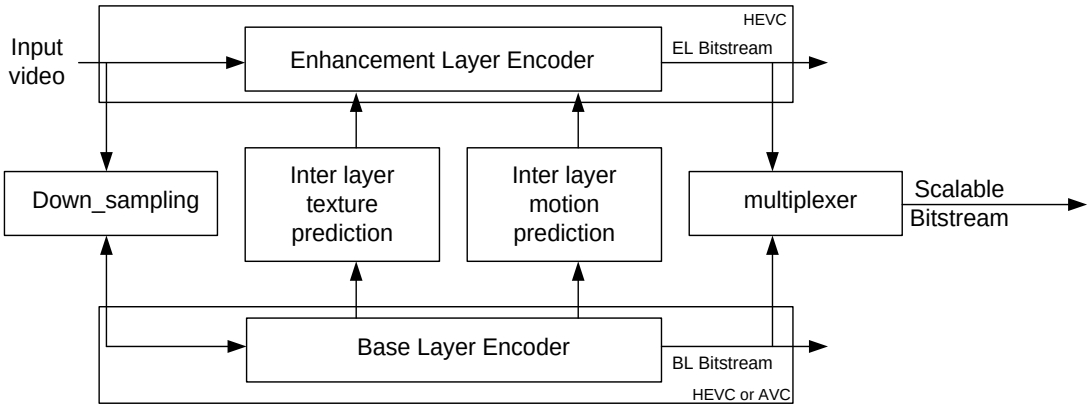



**Fig.2 Scalable scheme with two-layer structure**

In fact, a reduced quality version or lower resolution of each coded frame is classified as an ordinary HEVC codec in the Base Layer (BL). In addition to the HEVC predictions, Inter Layer Predictions (ILPs) is used for coding the Enhancement Layer with a higher resolution or quality.

ILPs consist of Inter Layer Motion Predictions ILMPs and Inter Layer Texture Predictions ILTPs. This module exploits information from lower resolutions. They could be either motion vectors to update the motion information or directional modes for texture information. SHEVC comprises three main types of video scalability: temporal, spatial and quality. The first helps to reduce the video frame rate. The second contributes to a decrease in spatial resolution. The third, considered as a special case of spatial scalability with identical picture sizes for base and ELs, lessens video details.

These three types of scalability were previously incorporated in SVC and used in SHVC. The new scalability types which emerged in SHVC were detailed in [11]. Since several works such as in [1] have been completed to prove their efficiency , these extinctions are worth investigating.

In order to improve SHVC time execution, an implementation of the original SHVC on TMS320C6678 is realized. Furthermore, an optimization of SHVC is also implemented in order to further improve video coding efficiency. These optimizations are explained in the next sub-sections.

### 2. SHVC OPTIMIZATION

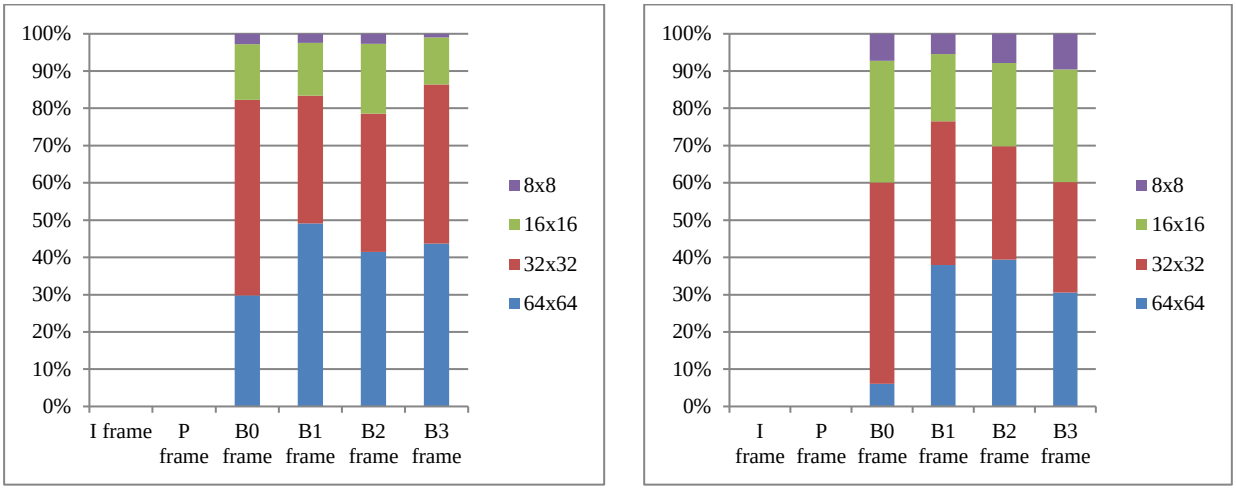
This optimization aimed to reduce the encoding process complexity. The proposed method was deeply detailed in earlier work [12].Computational time was reduced based on statistical studies elaborated in [1]. For simulations, all test sequences related to the two video classes were considered with different Quantization Parameters (QP) as shown in Table 1. Moreover, statistical analysis was performed for different block sizes. In addition, other test conditions, including frame number and coding configuration, are illustrated in Table 1 .

**Table 1. Test conditions**

| | |
|---|---|
| Number of frames taken for each video sequence | 100 |
| The quantization parameter pairs | {22, 24}, {34, 36} |
| The coding configuration | Random access |
| Video sequences classes | Class A, Class B |

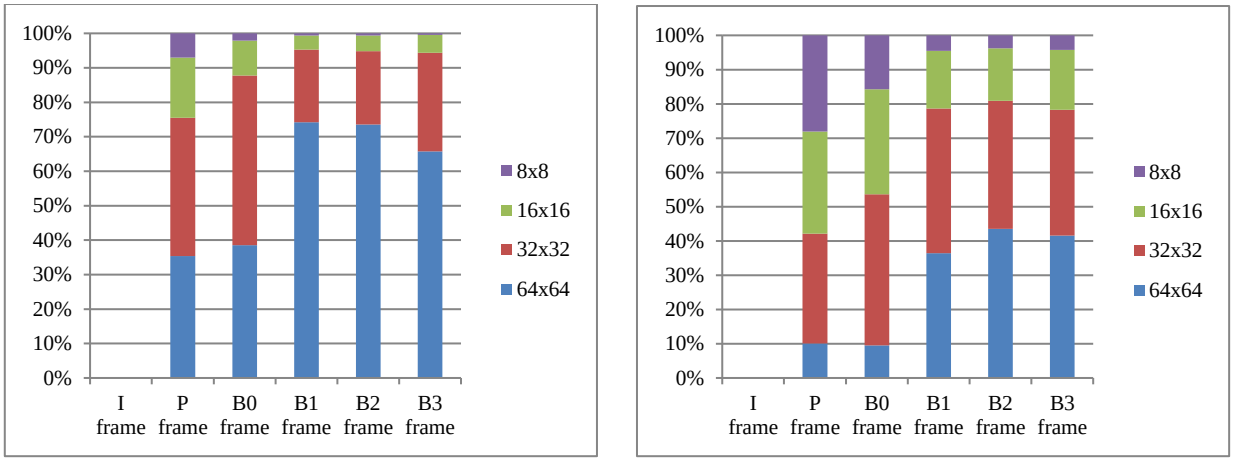
The use percentages concerning basketballdrive video sequence following block size and frame type at the CUs level is presented in Figure 3 and Figure 4. Results have shown that some CUs sizes were rarely or not used at all. This observation was made for BL and EL.


Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)




**(a) Inter for QPs {34,36}** **(b) Inter for QPs {22,24}**

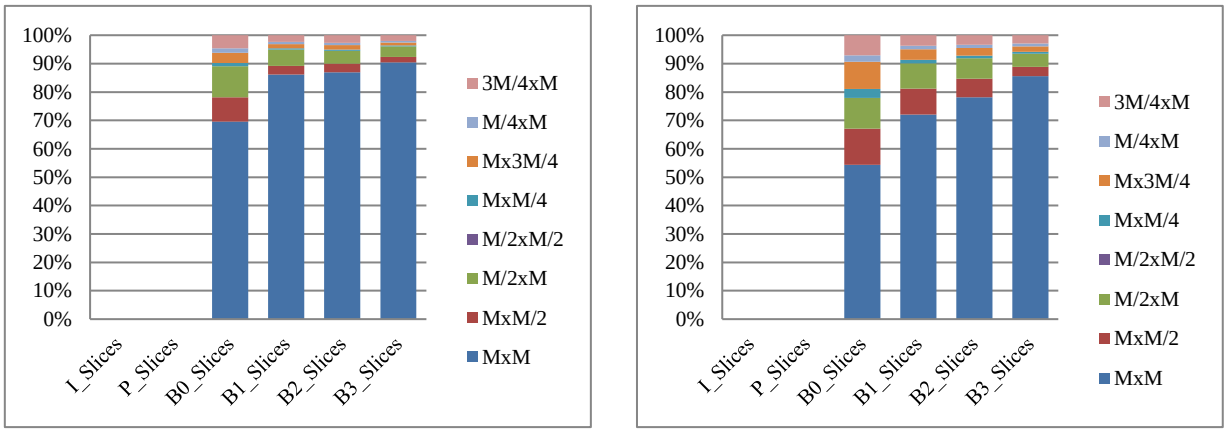
**Fig.3 Percentages of CU's size for inter-mode prediction for BL. (a) Inter for QPs{34,36}.(b) Inter for QPs{22,24}.**




**(a) Inter for QPs {34, 36}** **(b) Inter for QPs {22, 24}**

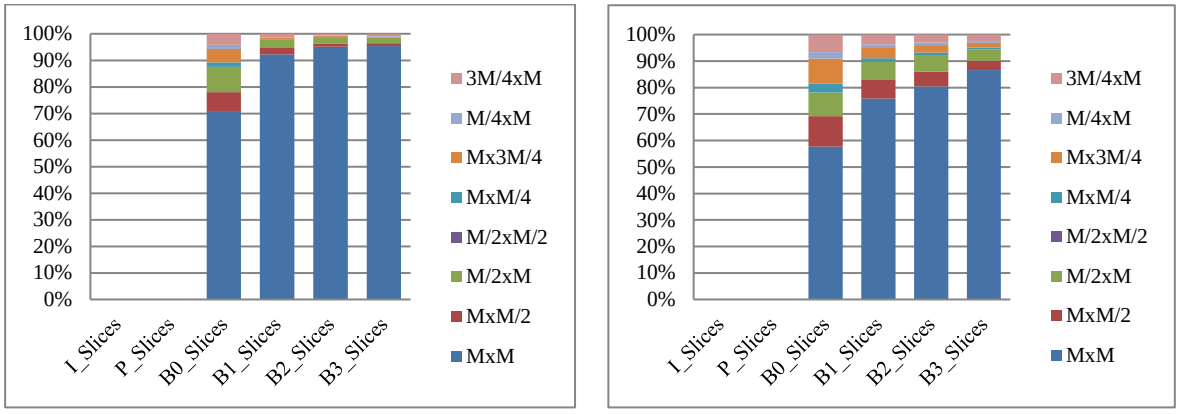
**Fig.4 Percentages of CU's size used for inter-mode prediction for the EL. (a) Inter for QPs {34, 36}. (b) Inter for QPs {22, 24}.**

Figure5 and Figure 6showa thoroughdistribution of PUs different sizes in the random access configuration mode for Basketballdrive video sequence at a QPs {34, 36} and QPs{22, 24} for BL and EL respectively. We alsonote that this distribution depends on QP values.




**(a) Inter for QPs {34,36}** **(b) Inter for QPs {22,24}**

**Fig.5 Percentages of depth for Inter-mode prediction at the PUs level in the BL. (a) QPs {34, 36}. (b) QPs {22, 24}.**




**(a) Inter for QPs {34,36}** **(b) Inter for QPs {22,24}**

**Fig.6 Percentages of depth for Inter-mode prediction at the PUs level in the EL. (a) QPs {34, 36}. (b) QPs {22, 24}.**

The proposed optimization intends to lessen computational effort depending on QP values and to avoid computations for unused sizes. Consequently, time execution is automatically saved.

The proposed algorithm is as follows:

- Step 1: identifies QP pair used for the simulations
- Step 2: as shown in statistical simulations, if QP pair is {22, 24}, the least used CUs size is8x8. As for the PUs type, the least used were the asymmetric ones for both BL and EL layers. Thus, computational process is interrupted, leading to the next CUs or PUs size.
- Step 3: If the QP pair is {34, 36}, the same strategy followed in step 2 is used; but with removing the computation for CU sized 16x16.
- Step 4: If the CUs size does not coincide with the ones less used, the original computational process will be conducted.

Two methods are investigated through the proposed algorithm. The first one focuses on the less used CUs while the second concerns the less used PUs. Athird method is a combination between the first method and the second one.

## V. SHVC IMPLEMENTATION

SHVC encoder implementation on TMS320C6678 processor was achieved to assess its performance. The DSP core clock frequency was set to 1.25 GHZ. Moreover, no Operating System running at the processor level was included. In fact, the DSP core executed its own instance at a Real Time Operating System (RTOS) called SYSBIOS [13]. SHVC encoder was performed using Integrated Development Environment (IDE) Code Composer Studio and SHM6 as a SHVC reference software [15].A RTOS task relating to the encoding process was also implemented. It is composed of an Input file Processing process, an encoding process and two output files processing process. Input file processing module reads the input video file. This input file is then introduced to the encoding process for compression. At the end, Output File Processing process stores the reconstructed video and the bitstream on two different files. In order to optimize the SHVC encoder performance, the SYSBIOS operating system running on the DSP core was configured. Moreover, L1 and L2 internal memories were configured as cache memories at their maximum size. For L1 and L2 memories, we have fixed 32KB for both data and program were loaded using the internal memories. RTOS task has a stack of 20 MB and a heap of 4 MB. The external memory DDR3 is cacheable. Moreover, SHM6 the C/C++ software code was adapted with the SYSBIOS RTOS to finally obtain a fully-functional running encoder on the DSP core.

## VI. EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the original SHVC implementation and optimized version are evaluated through simulation studies. Our simulations are carried out on anEVM6678 [14]. This board serves as a hardware reference design for TMS320C6678 DSP. It has 512 MB of DDR3 RAM memory available for image storage. The number of frames taken in each sequence is 50. In fact, the original SHVC reference software algorithm SHM6.0 [15] was implemented on a single core C6678 running at 1.25GHz. In addition, the optimized algorithms presented in this paper were implemented onSHM6.0 reference software [15] and simulations were also carried out on EVM6678. In order to assess the coding performance and computational complexity of the original and the proposed algorithm, different experiments were carried out using all test video sequences. The tested video sequences were encoded using two QP pairs: {22, 24} and {34, 36}. In addition, the coding configuration was fixed as Main configuration based on the common JCT-VC test conditions [16].The percentage difference in Cycles Numbers (ΔNC) was used to compare the computation complexity of our proposed algorithms with the original SHM6 reference software. The presented criterion was calculated using Eq. 1.

$$\Delta NC = \frac{NC_p - NC_O}{NC_O} \times 100 \quad (1)$$

Where NCp represents the number of cycles when encoding the proposed optimization algorithm and NCo denotes the number of cycles when encoding the original algorithm. The coding efficiency of our proposed algorithm are evaluated using the Bjontegaard metrics as Bjontegaard peak signal-to-noise ratio (BD-PSNR) and Bjontegaard bitrate (BD-BR) [17]. The results are summarized in Table 2 which shows the encoding comparison results between the original algorithm and our proposed optimizations where significant gain is observed.



Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

**Table2. CUs method performance in terms of biterate, quality and number of cycles**

| Video sequences | QP values | CUs | | |
|---|---|---|---|---|
| | | BD-BR (%) | BD-PSNR (br) | Gain on cycle's number using the first optimization method (%) |
| Basketballdrive | 22, 24 | 1,07 | 0,010 | 39% |
| | 34, 36 | 1,72 | -0,004 | 26% |
| BQTerrace | 22, 24 | 0,62 | -0,020 | 19% |
| | 34, 36 | 0,7 | -0,001 | 17% |
| Cactus | 22, 24 | 0.00 | 0.000 | 31% |
| | 34, 36 | 0.00 | 0.000 | 28% |
| Kimono | 22, 24 | 0,10 | -0,019 | 28% |
| | 34, 36 | -0,24 | -0,010 | 24% |
| Parkscene | 22, 24 | 0,80 | -0,030 | 25% |
| | 34, 36 | 0,40 | 0,010 | 20% |
| **Average ClassB** | **22, 24** | **0,52** | **-0,012** | **28%** |
| | **34, 36** | **0,52** | **-0,001** | **23%** |
| PeopleOnSteet | 22, 24 | 1.00 | -0,009 | 30% |
| | 34, 36 | 0,90 | 0,040 | 26% |
| Traffic | 22, 24 | 1,35 | -0,007 | 21% |
| | 34, 36 | 2,24 | -0,009 | 18% |
| **Average Class A** | **22, 24** | **1,18** | **-0,008** | **25%** |
| | **34, 36** | **1,57** | **0,016** | **22%** |
| **Average** | **22, 24** | **0,85** | **-0,010** | **27%** |
| | **34, 36** | **1,04** | **0,007** | **22%** |

The CUs method implementation shown in Table 2provides an interesting gain on cycles' number reaching an average up to 27%, with an insignificant loss in quality and a negligible rise in bit-rate. The best result was obtained for Basketballdrive video sequence with a percentage of 39% in terms of an execution cycle account reduction with an acceptable quality drop and a bit-rate rise. We may also notice that, in average, and for both test video classes, results were close to 25% for both QP pairs.

**Table 3. PUs and Combined CUs-PUs method performance in terms of biterate, quality and number of cycles**

| video sequences | QP values | PUs | | | CUs-PUs | | |
|---|---|---|---|---|---|---|---|
| | | BD-BR (%) | BD-PSNR (BR) | Gain on cycle's number using the second optimization method (%) | BD-BR (%) | BD-PSNR (BR) | Gain on cycle's number using the combined optimization method (%) |
| Basketballdrive | 22, 24 | 0,94 | -0,020 | 20% | 2,37 | -0,005 | 51% |
| | 34, 36 | -0,50 | 0,007 | 15% | 2,05 | 0,000 | 38% |
| BQTerrace | 22, 24 | 0,50 | -0,001 | 15% | 0,80 | -0,05 | 35% |
| | 34, 36 | 0,25 | -0,010 | 7% | 1,33 | -0,008 | 20% |
| Cactus | 22, 24 | 0.00 | 0.000 | 3% | 0.00 | 0.000 | 33% |
| | 34, 36 | 0.00 | 0.000 | 9% | 0.00 | 0.000 | 33% |
| Kimono | 22, 24 | 1,50 | 0,010 | 12% | 2.00 | -0,02 | 40% |
| | 34, 36 | 1,03 | 0,006 | 10% | 1,32 | -0,015 | 35% |
| Parkscene | 22, 24 | 0.52 | 0,012 | 11% | 1,23 | -0,050 | 32% |
| | 34, 36 | -0,40 | 0,008 | 9% | 0,28 | 0.000 | 30% |
| **Average ClassB** | **22, 24** | **0,63** | **0,000** | **12%** | **1,28** | **-0,025** | **38%** |
| | **34, 36** | **0,08** | **0,002** | **10%** | **0,98** | **-0,005** | **31%** |
| PeopleOnSteet | 22, 24 | 1,20 | -0,005 | 20% | 2,50 | -0,010 | 45% |
| | 34, 36 | 0,51 | 0,009 | 19% | 1,53 | 0,010 | 41% |
| Traffic | 22, 24 | 0,50 | 0.000 | 9% | 2.00 | -0,009 | 43% |
| | 34, 36 | 0,90 | 0.000 | 6% | 3,10 | -0,010 | 33% |
| **Average ClassA** | **22, 24** | **0,85** | **-0,003** | **15%** | **2,25** | **-0,010** | **44%** |
| | **34, 36** | **0,71** | **0,005** | **12%** | **2,32** | **0.000** | **37%** |
| **Average** | **22, 24** | **0,74** | **-0,001** | **13%** | **1,77** | **-0,017** | **41%** |
| | **34, 36** | **0,39** | **0,003** | **11%** | **1,65** | **-0,002** | **34%** |

The PU implementation method depicted in Table 3 displays good results reaching, in average,13% and 11% for the two QP pairs. This result is associated with a negligible increment in bit-rate and an acceptable loss in quality. A combination of the two methods is implemented and their results are presented in Table 3. An important gain in terms of cycles' number reduction is noted, achieving 41% for QP {22, 24} and 34% for QP {34, 36} in average. These achievements are accompanied with a tolerable loss in quality of about -0.017 and -0.002 and an acceptable increase of about 1.77 and 1.65 in terms of bit-rate for both pairs of QPs respectively. Speed improvements obtained from the test video sequences indicate a consistent contribution of the proposed algorithmic optimization. Compared to the previous studies, detailed in Table 4, our proposed implementation on DSP TMS320C6678proves to be able to achieve the best coding speeds while using equivalent hardware resources. We also find out that our implementation provides the best result with a percentage gain of about 41%. Unlike the other state of the art studies which focus on the decoder side more easily implemented, our work touches the encoder side challenging its complexity and implementation difficulty.

**Table 4. Comparison with the state of the art methods**

| Encoder/ Decoder | Encoder/Proposed methods | [7](Decoder) | [9](decoder) |
|---|---|---|---|
| Platform | DSP-TMS320C6678 | GPAC multimedia library into MP4 file format | 6 cores Xeon processor |
| Speed performance (percentage of cycles' number) | 41% | 27% | 31% |

## VII. CONCLUSION AND PERSPECTIVE

The scalable high video coding is an important solution for video transmission over heterogeneous networks. In order to improve real time video coding, an implementation on DSP was carried out. SHVC video encoder optimization on a multicore DSP TMS320C6678 was achieved. Indeed, algorithmic optimizations were proposed then implemented on one core. The obtained encoding speed was up to 41% in average. Therefore, our implementation obeyed the real-time application constraint without causing a significant drop in the quality or a major increase in terms of bit-rate. In future work, we intend to propose a frame level parallelism implementation targeting the use of the entire cores in TMS320C6678 as well as an enhanced direct memory access (EDMA) controller being the backbone of the two-level cache architecture for TMS320C66x which can be a great solution for data transfer that needs to be further investigated.

## REFERENCES

1. Heiko Schwarz. Detlev Marpe. Thomas WiegandOverview of the scalable video coding extension of the H.264/AVC standardIEEE Trans. Circuits Syst. Video Technol.. 17 (9) (2007). pp. 1103–1120
2. Wali, I., Kessentini, A., Ayed, M. A. B., & Masmoudi, N. (2017). Depth partitioning and coding mode selection statistical analysis for SHVC. International Journal of Advanced Computer Science and Applications (IJACSA), 8(1).
3. Shirvaikar, M., & Bushnaq, T. (2009, February). A comparison between DSP and FPGA platforms for real-time imaging applications. In Real-Time Image and Video Processing 2009 (Vol. 7244, p. 724406). International Society for Optics and Photonics.
4. Pescador, F., Juarez, E., Raulet, M., & Sanz, C. (2011). A DSP based H. 264/SVCdecoder for a multimedia terminal. IEEE Transactions on Consumer Electronics, 57(2), 705-712.
5. Pescador, F., Daian, A., Fernández, I., Juárez, E., & Garrido, M. (2013, June). Multi-DSP implementation of a H. 264/SVC decoder. In 2013 IEEE International Symposium on Consumer Electronics (ISCE) (pp. 145-146). IEEE.
6. Bahri, N., El Ansari, A., Maazouz, M., Ahaitouf, A., & Masmoudi, N. (2016, December). HEVC video encoder implementation on Texas Instruments platforms. In 2016 17th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA) (pp. 165-170). IEEE.
7. Kibeya, H., Bahri, N., Ayed, M. A. B., & Masmoudi, N. (2016, November). SAD and SSE implementation for HEVC encoder on DSP TMS320C6678. In 2016 International Image Processing, Applications and Systems (IPAS) (pp. 1-6). IEEE.
8. Hamidouche, W., Raulet, M., & Deforges, O. (2014, October). Real time shvc decoder: Implementation and complexity analysis.
9. In 2014 IEEE International Conference on Image Processing (ICIP) (pp. 2125-2129). IEEE.
10. Hamidouche, W., Cocherel, G., Le Feuvre, J., Raulet, M., & Deforges, O. (2014, July). 4K real time video streaming with SHVC decoder and GPAC player. In 2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW) (pp. 1-2). IEEE.
11. Hamidouche, W., Raulet, M., & Déforges, O. (2014, July). Parallel SHVC decoder: Implementation and analysis. In 2014 IEEE International Conference on Multimedia and Expo (ICME)(pp. 1-6). IEEE.
12. Chen, J., Boyce, J., Yan, Y., Hannuksela, M.M., Sullivan, G.J., Wang, Y.-K.: Scalable High Efficiency Video Coding Draft 7, JCTVC-R1008. In: 18th JCTVC Meeting, Sapporo (2014)
13. Wali, I., Kessentini, A., Ayed, M. A. B., & Masmoudi, N. (2019). Fast inter-prediction algorithms for spatial Scalable High efficiency Video Coding SHVC. Signal, Image and Video Processing, 13(1), 145-153.
14. TI SYS/BIOS v6.45 Real-time Operating System User's Guide. http://www.ti.com/lit/pdf/spruex3.pdf.
15. TMS320C6678 Multicore Fixed and Floating-Point Digital Signal Processor. http://www.ti.com/lit/ds/sprs691c/sprs691c.pdf, February 2012. Texas Instruments Literature Number: SPRS691C
16. Scalable HEVC (SHVC) Test Model 6 (SHM 6). https://hevc.hhi.fraunhofer.de/svn/svn_SHVCSoftware/tags/SHM-6.0/
17. Seregin, V., He, Y.: Common SHM test conditions and software reference configurations. In: Document JCTVC-O1009. Geneva (2013)
18. Bjontegaard, G.:Calculation of average PSNR differences between RD-curves Doc. VCEG-M33. Austin (2001)

## AUTHOR PROFILE

**Ibtissem Wali,** received her degree in electrical ingeneering in 2013 from Sfax National School of Engineering (ENIS), Tunisia and her Master degree from the Lorraine University in 2014, France. She obtained her doctor degree in 2018 from the Sfax National School of Engineering (ENIS).

**Amina Kessentini,** was born in Sfax, Tunisia, in 1983. She received her degree in Electrical Engineering and her Masters in Electronic Engineering from Sfax National School of Engineering (ENIS), Tunisia, in 2007 and 2008, respectively. In 2015, she joined the Higher Institute of Computing and Multimedia of Gabes, Tunisia, as an assistant professor. She is a member of Sfax Laboratory of Electronics and Information Technology. Her current research interests include video coding and development of video compression algorithms.



Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

**Mohamed Ali Ben Ayed,** wasborn in Sfax, Tunisia, in 1966.He received his BS degree inComputer Engineering fromOregon State university and MSdegree in Electrical Engineeringfrom Georgia Institute ofTechnology in 1988, his DEA,Ph.D., and HDR degrees inElectronics Engineering fromSfax National School ofEngineering in 1998, 2004, and2008, respectively. He is currentlya professor in theDepartment of Communicationat Sfax High Institute of Electronics and Communication. He was a cofounderof BUbvideo Tunisia in the techno-pole El- GHAZLATunis, aninternational leader company in the domain of video coding technology.He is a member of a research team since 1994 at (LETI, Sfax) in thedomain of Electronics and Information Technology and a reviewer inmany international and national journals and conferences. He is currentlya technical advisor of BEBREASK video^, a Research and Developmentcompany specialized on the next high-efficient video coding generationH265. His current research interests include DSP and VHDL implementationof digital algorithms for multimedia services, and development ofdigital video compression algorithms.

**Nouri Masmoudi,** received electrical engineering degree from theFaculty of Sciences andTechniques—Sfax, Tunisia, in1982, the DEA degree from theNational Institute of AppliedSciences—Lyon and UniversityClaude Bernard—Lyon, Francein 1984. From 1986 to 1990, heprepared his thesis at the laboratoryof Power Electronics (LEP) atthe National School of Engineers of sfax (ENIS). He received hisPh.D. degree from the NationalSchool Engineering of Tunis(ENIT), Tunisia in 1990. From 1990 to 2000, he is currently a professorat the electrical engineering department—ENIS. Since 2000, he has beena group leader ‘Circuits and Systems’ in the Laboratory of Electronics andInformation Technology. Since 2003, he is responsible for the ElectronicMaster Program at ENIS. His research activities have been devoted toseveral topics: Design, Telecommunication, Embedded Systems,Information Technology, Video Coding and Image Processing.