SYSTEM ANALYSIS OF VLSI ARCHITECTURE FOR MOTION-COMPENSATED TEMPORAL FILTERING

Ching-Yeh Chen, Chao-Tsung Huang, Yi-Hau Chen, Chung-Jr Lian, and Liang-Gee Chen

DSP/IC Design Lab, Graduate Institute of Electronics Engineering and
Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan
Email:{cychen, cthuang, ttchen, cjlian, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

The Motion-Compensated Temporal Filtering (MCTF) is an innovative prediction scheme for video coding and has become the core technology of the coming video coding standard, MPEG-21 part 13 - Scalable Video Coding (SVC). This paper provides the system analysis of MCTF for VLSI implementation, which includes computational complexity, external memory access, external storage size, and coding delay. The one-level MCTF is analyzed first, and a modified double current frames scheme will be introduced to address the external memory access penalty that results from fractional-pel Motion Compensation (MC). Then the analysis is extended to multi-level MCTF, in which many important system issues will be explored. Finally, a real-life test case will be given to compare the system requirements of many different MCTF schemes and the prediction scheme of H.264/AVC.

1. INTRODUCTION

In this decade, the open-loop MCTF prediction scheme has been developed to enable efficient scalable video coding, in which the concept is to perform a wavelet transform in the temporal direction with MC. For more details, please refer to [1]. MPEG has identified a set of applications that require scalable and reliable video coding technologies. After evaluating the response to Call for Proposals on Scalable Video Coding (SVC) [2], it has been shown that there is a new and innovative video technology that MPEG can bring to industry in a future video standard. Currently, the scalable extension of H.264/AVC with MCTF is adopted as Scalable Video Model (SVM) 3.0 [3]. The lifting-based MCTF is the core technology to provide scalable video coding. The MCTF not only can provide a variety of efficient scalabilities because the drift problem of traditional close-loop prediction scheme is prevented by the open-loop structure but also can increase the compression efficiency of H.264/AVC [4].

In our previous work [5], the one-level MCTF with 5/3 or 1/3 filter is analyzed, but only integer-pel MC is considered. In this paper, we will consider the impact of fractional-pel MC and the bi-iterative motion vector refinement for one-level MCTF. A modified double current frames scheme is proposed to reduce the penalty of fractional-pel MC. The multi-level MCTF will be analyzed in detail to show the trade-offs among different MCTF schemes, including computation complexity, external memory access, external storage size, and coding delay. Then a practical test case will be used to evaluate the system issues of MCTF schemes and compare them with H.264/AVC prediction scheme. This paper is organized as follows. In section 2, the MCTF schemes using the Haar, 1/3, and 5/3 filters, are introduced. The analysis of one-level MCTF and multi-level MCTF is presented in sections 3 and 4, respectively. Section 5 will conclude this paper.

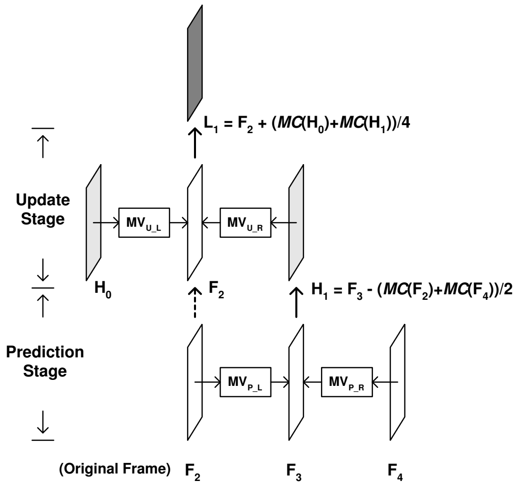

Fig. 1. The 5/3 MCTF scheme. MV_{P_L} and MV_{P_R} represent the motion vectors from the left and right neighbor frames for the prediction stage, respectively, and so represent MV_{U_L} and MV_{U_R} for the update stage. The light gray frames (H) are the highpass frames, and the heavy gray frames (L) are the lowpass frames.

2. MOTION-COMPENSATED TEMPORAL FILTERING

MCTF is to perform wavelet transform in the temporal direction with MC. The coding performance and coding delay depend on which wavelet filter is adopted. From recent experimental results [1, 4], the MCTF is usually implemented by use of the 5/3, 1/3, or Haar filter with the lifting scheme. The 5/3 MCTF can achieve better coding performance than the 1/3 MCTF but requires longer encoding delay [6]. The lifting scheme is an efficient implementation method of wavelet filters, and it can guarantee the perfect reconstruction property. For simplicity, MCTF represents the lifting-based MCTF using the 5/3 or 1/3 filter in the following.

The 5/3 MCTF can be simply illustrated by Fig. 1, in which only two lifting stages are involved. The prediction stage is using even frames to predict odd frames, and the residual frames are the highpass frames. The update stage is using the highpass frames

---

This work was supported in part by National Science Council, Republic of China, under the grant number 91-2215-E-002-035 and in part by the MediaTek Fellowship.

0-7803-9134-9/05/$20.00 ©2005 IEEE

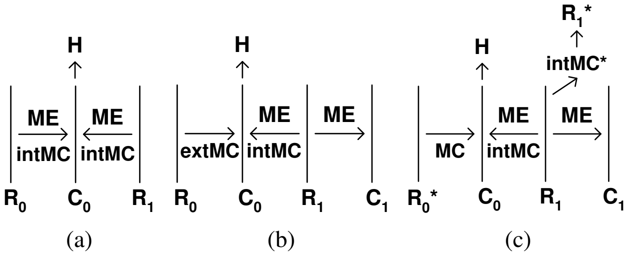

Fig. 2. Data reuse schemes for the prediction stage. (C: Current frame; R: Reference frame.) (a) Double reference frames ME; (b) Double current frames ME; (c) Modified double current frames ME.

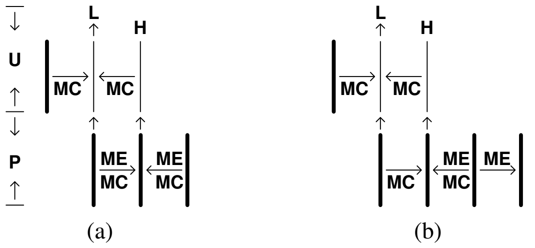

Fig. 3. Frame-level data reuse schemes of 5/3 MCTF. (P: Prediction stage; U: Update stage.) (a) P-DRF/U-DRF; (b) P-DCF/U-DRF.

to update the even frames, and the derived frames are the lowpass frames. The 1/3 MCTF is just to skip the update stage of the 5/3 MCTF and treat the even frames as the lowpass frames. For aligning the objects in different frames, the two lifting stages both require motion vectors. The block-based motion model is usually adopted. For every block in the odd frames, ME should be performed to find the best motion vectors MV_{P_L} and MV_{P_R} in the prediction stage. These two motion vectors can be refined in a bi-iterative way like SVM3.0. As for the update motion vectors MV_{U_L} and MV_{U_R}, they are usually estimated and derived from MV_{P_L} and MV_{P_R} for saving the motion vector cost.

Fig. 1 only shows the one-level MCTF scheme. The multi-level MCTF can be derived by recursively performing one-level MCTF on the L-frames. The multi-level 5/3 and 1/3 MCTF are performed in a bottom-to-top order that is from higher to lower frame rates. In SVM3.0, a coding scheme called Hierarchical B-frames (HB) is introduced to provide an H.264/AVC compatible scalable coding bitstream. HB is to perform multi-level MCTF in a top-to-bottom way to be compatible with the generic B-frames of H.264/AVC. The coding performance of HB is very similar with 1/3 MCTF in SVM3.0.

3. ANALYSIS OF ONE-LEVEL MCTF

3.1. Prediction Stage

The Motion Estimation (ME) is the most computation consuming operation in the prediction stage. There are many macroblock-level data reuse schemes for ME, and the redundancy access factor (Ra) is used to evaluate the external memory bandwidth [7]. Ra can be defined as:

$$Ra = \frac{memory\ bandwidth\ for\ reference\ frame}{minimum\ memory\ bandwidth}$$

which means the redundant memory access for reference frames. In [5], we propose two frame-level data reuse schemes for prediction stages, which are Double Reference Frames (DRF) and Double Current Frames (DCF) as Fig. 2(a) and (b), respectively. DRF performs the bi-directional ME for every current block together, while DCF is proposed to reduce the memory bandwidth by sharing the search range buffer for two current blocks. However, the memory bandwidth analysis of [5] is only for integer-pel MC without bi-iterative motion vector refinement.

The MC can be further categorized into internal MC (intMC) and external MC (extMC). In Fig. 2(a), DRF can perform the MC internally without external memory access because the search range buffer has sufficient data for fractional-pel MC. The left branch of MC in Fig. 2(b) needs external memory access to perform MC. If the motion vector is fractional-pel, more data is required from external memory access to interpolate the fractional-pel MC block. Moreover, if the bi-iterative refinement is required, the data in the refinement search range should also be read from the external memory. For extMC of DCF with bi-iterative refinement, the total amount of data required from external memory, A, is $(B_{MC,H} + SR_{Bi} + F - 1)(B_{MC,V} + SR_{Bi} + F - 1)$, where $B_{MC,H}$ and $B_{MC,V}$ are the block width and height, F is the interpolation filter length, and SR_{Bi} is the refinement search range for less than four times of bi-iterations. The corresponding redundancy access factor, $Ra_{extMC-Bi}$, can be defined by $Ra_{extMC-Bi} = \frac{A - B_{MC,H} \times B_{MC,V}}{B_{MC,H} \times B_{MC,V}}$, which depends on the block size. For SVM3.0, the typical values of $Ra_{extMC-Bi}$ are listed in Table 1. For small blocks, the bandwidth overhead of extMC is quite large. The bi-iterative refinement makes the overhead larger. The memory access of DCF becomes $(Ra + \overline{Ra_{extMC-Bi}})/2 + 2$, but that of DRF is still $Ra+1$. Although DCF can reduce the bandwidth of ME, it suffers the bandwidth overhead of extMC, which depends on the average $\overline{Ra_{extMC-Bi}}$ of all blocks. Moreover, the overhead of extMC could make DCF less efficient than DRF if $\overline{Ra_{extMC-Bi}}$ is larger than $Ra - 2$, which happens very possibly.

If the bi-iterative refinement is not performed (SR_{Bi}=0), the overhead of extMC is still large. We propose to use the modified DCF (m-DCF) in Fig. 2(c) to reduce the bandwidth by interpolating the best matched blocks of R_1 to R_1^* for C_1 in advance and saving them into external memory. The m-DCF can reduce memory access to $Ra/2 + 2.5$. Moreover, the memory access of R_0^* is very regular compared to that of R_0 in DCF.

3.2. Combined Prediction and Update Stage

In the update stage, only the extMC is performed, and the motion vectors are derived from those in the prediction stage. It has been shown that it is better to perform MC of the update stage using the DRF scheme [5]. The memory access of DRF scheme in the update stage can be found to be $\overline{Ra_{extMC}} + 2$ where $Ra_{extMC} =$

Table 1. List of $Ra_{extMC-Bi}$ for SVM3.0 where H.264/AVC 6-tap interpolation filter is adopted. Bi is the number of bi-iteration times.

MC Block Type	4x4 Block (B_H=B_V=4)	8x8 Block (B_H=B_V=8)	16x16 Block (B_H=B_V=16)
$Ra_{extMC-Bi}$ (SR_{bi}=0 or Bi<=1)	4.0625	1.6406	0.7227
$Ra_{extMC-Bi}$ (SR_{bi}=2 and Bi=2,3)	6.5625	2.5156	1.0664

Table 2. Summary of external memory access and external storage size for one-level MCTF (Bw: External memory bandwidth; EMS: External Memory Storage)

	5/3 MCTF			**1/3 MCTF**		
Prediction	**DRF**	**DCF**	**m-DCF**	**DRF**	**DCF**	**m-DCF**
Update	**DRF**	**DRF**	**DRF**	**-**	**-**	**-**
Bw (pixels/pixel)	$Ra+3+\overline{Ra}_{extMC}$	$(Ra+\overline{Ra}_{extMC-Bi})/2+\overline{Ra}_{extMC}+4$	$Ra/2+4.5+\overline{Ra}_{extMC}$	$Ra+1$	$(Ra+\overline{Ra}_{extMC-Bi})/2+2$	$Ra/2+2.5$
EMS (frames)	**4**	**5**	**6**	**3**	**4**	**5**

$Ra_{extMC-Bi|Bi=0}$. In SVM3.0, the blocks in the update stage are all 4×4 that makes the extMC bandwidth overhead very heavy. Fig. 3 shows the frame-level reuse schemes for 5/3 MCTF. The frames expressed by bold lines represent those need to be stored in the external memory for performing 5/3 MCTF. If m-DCF is used in Fig. 3(b), one more frame will be required to be stored in the external memory. The analysis results for one-level MCTF are summarized in Table 2 where the average bandwidth is used. For the worst case of memory bandwidth for SVM3.0, the maximum value of $Ra_{extMC-Bi}$ in Table 1 should be used.

4. ANALYSIS OF MULTI-LEVEL MCTF

When extending the one-level MCTF to multi-level MCTF, three preconditions should be given. The first one is how many levels should be performed. According to the coding results using SVM3.0, four-level MCTF has the best compression efficiency for CIF sequences $Mobile\&Calendar$ and $Foreman$, but two-level MCTF is the best for the sequence $Stefan$. It means different levels of MCTF could be used for sequences of different characteristics. The second precondition is to perform inter coding for the lowpass frames (L-frames) or not. The third one is to perform 1/3 MCTF or HB if the update stage is not performed. In the following analysis, MCTF uses the closest frame as the reference frame of each direction for the prediction stage, and the L-frames are inter-coded as IPPPP.. structure using M previous frames as reference frames. Since the hardware requirements of open-loop 1/3 MCTF and HB are all exactly the same, HB will not be mentioned in the below.

4.1. Computation complexity and External Memory Access

Different MCTF levels have different computation complexity and external memory access. The redundancy access factors in each MCTF level can be different. In the following, the redundancy access factors are assumed to be the same for every MCTF level. Then computation complexity and external memory access are exponentially decreased for higher MCTF levels. As shown in Fig. 4, the input frames in the second level MCTF is one half of those in the first level MCTF, and so as the workload (WL). If the workloads are assumed to be dominated by ME and MC, WL can be formulated as follows:

$$WL_{1^{st}-level} = \begin{cases} 1ME + 2MC \ for \ 5/3MCTF \\ 1ME + 1MC \ for \ 1/3MCTF \end{cases}$$

$$WL_L = \frac{M \times ME + 1MC}{2^J}$$

$$WL_{J-level} = (1 + \frac{1}{2} + ... + \frac{1}{2^{J-1}})WL_{1^{st}-level} + WL_L$$

where $WL_{1^{st}-level}$, WL_L, and $WL_{J-level}$ are average workloads per frame for the first level MCTF, inter-coded L-frames, and J-level MCTF, respectively. It can be found that the computation complexity of J-level MCTF is very close to traditional close-loop MC prediction with two reference frames.

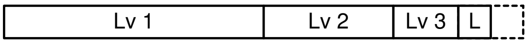

Fig. 4. Scaling effect of workload for three-level MCTF and inter-coded L-frames with 1-ref frame.

Table 3. Coding delays of different data reuse schemes

Data Reuse Scheme	**5/3 MCTF P-DRF/U-DRF**	**5/3 MCTF P-DCF/U-DRF**	**1/3 MCTF P-DRF**	**1/3 MCTF P-DCF**
Coding Delay (frames)	$3(2^J-1)$	$4(2^J-1)$	2^J-1	$2^J+2^{J-1}-1$

The external memory bandwidth (Bw) has similar scaling effect as computation complexity:

$$Bw_{J-level} = (1 + \frac{1}{2} + ... + \frac{1}{2^{J-1}})Bw_{1^{st}-level} + \frac{1 + M \cdot Ra}{2^J}$$

where $Bw_{1^{st}-level}$ is as shown in Table 2. $Bw_{J-level}$ is close to the double of $Bw_{1^{st}-level}$ if $M \leq 2$.

4.2. External Memory Storage

The required external memory storage (EMS) of MCTF is linearly proportional to J. For 5/3 MCTF,

$$EMS_{J-level,5/3} = J \cdot EMS_{1^{st}-level,5/3} + M.$$

where $EMS_{1^{st}-level,5/3}$ is as shown in Table 2. For 1/3 MCTF, the frame R_0 in Fig. 2 can be shared among all MCTF levels because no update stages are performed. Thus, for 1/3 MCTF,

$$EMS_{J-level,1/3} = J(EMS_{1^{st}-level,1/3} - 1) + M.$$

where $EMS_{1^{st}-level,1/3}$ is as shown in Table 2. The EMS of multi-level MCTF is much larger than the traditional close-loop MC prediction when J is large.

4.3. Coding Delay

The coding delay is an important issue for the open-loop MCTF prediction because it is much longer than traditional MC prediction. In [6], only the encoding delay is discussed. In the following, the coding delay is considered, which is defined as the maximum distance between the decoded frame, say X, and the farthest frame that is required to encode frame X. Fig. 5 is an example of two-level 5/3 MCTF system. The coding delay of P-DRF/U-DRF scheme is 9 frames, and that of P-DCF/U-DRF scheme is 12.

The coding delays for different data reuse schemes are formulated as Table 3. The coding delays of P-DRF/U-DRF and P-DRF scheme can be derived by using multi-rate filter bank equations. The encoding delay of P-DRF/U-DRF scheme is shown to be $2(2^J - 1)$ frames [6]. Similarly, the encoding delay of P-DCF/U-DRF scheme can be proven to be $3(2^J - 1)$ frames. Thus, the coding delay of P-DCF/U-DRF scheme can be derived because it happens that the coding delay path is the sum of longest delay

Table 4. System requirement comparisons of 4-level MCTF and H.264/AVC for CIF 30fps sequence

	5/3 MCTF			1/3 MCTF and HB			H.264 IBBPBBP	H.264 IBPBP
Prediction	DRF	DCF	m-DCF	DRF	DCF	m-DCF	DRF	DRF
Update	DRF	DRF	DRF	-	-	-	-	-
Bw (MB/sec)	69.9257	80.0830	64.2233	35.3549	45.5121	29.6525	23.3165	33.4541
EMS (MB)	1.7234	2.1289	2.5344	0.9124	1.3179	1.7234	0.4055	0.3041
Coding Delay (sec)	1.5000	2.0000	2.0000	0.5000	0.7667	0.7667	0.0667	0.0333

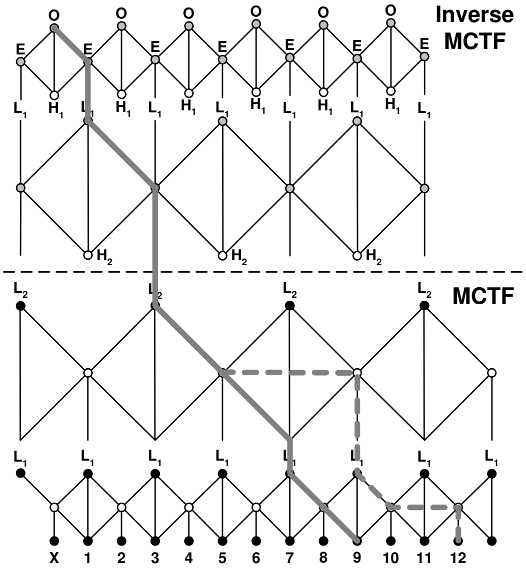

Fig. 5. Signal flow graph of two-level 5/3 MCTF system. (Solid line: P-DRF/U-DRF; Dot line: P-DCF/U-DRF)

paths in MCTF and inverse MCTF for 5/3 MCTF. However, the coincidence does not happen to 1/3 MCTF. Instead, the coding delay of P-DCF scheme can be derived from the signal flow graph of 1/3 MCTF. In summary, the coding delays of multi-level MCTF are exponentially increased with J. The ratio of coding delays for 5/3 P-DRF/U-DRF, 5/3 P-DCF/U-DRF, 1/3 P-DRF, and 1/3 P-DCF is about 3:4:1:1.5.

4.4. Summary of Multi-level MCTF

The computation complexity is very similar for all kinds of configurations for multi-level MCTF. The external memory bandwidth depends on the frame-level data reuse schemes, but is quite similar for different MCTF levels if the frame-level data reuse scheme is fixed. However, the external memory storage requirement is linearly proportional to the MCTF level, and the coding delay is exponentially increased as the MCTF level.

4.5. Case Study

To show the real-life system requirement, a case study is given for four-level MCTF for CIF 30fps sequences. The search range of ME is [-32,31], and ME is performed on a 16×16 macroblock basis. Level C data reuse scheme is adopted such that $Ra = 5$. Assume the extMC is all performed on a 4×4 block such that $Ra_{extMC} = 4.0625$. And $Ra_{extMC-Bi} = 6.5625$ if bi-iterative refinement is required for the DCF scheme of the prediction stage. The L-frames are inter-coded as IPPPP.. structure with one reference frame. Two configurations of H.264/AVC, IBBPBBP and IBPBP with two reference frames, are also compared (only the MC prediction). The comparisons are listed in Table 4. The m-DCF scheme can reduce the external memory access but requires more external storage and coding delay. Compared to H.264/AVC configurations, 1/3 MCTF and 5/3 MCTF require similar and double of the external memory access, respectively. The external storage requirement of MCTF is several times of that of H.264/AVC. Moreover, the coding delay of MCTF is more than an order of that of H.264/AVC.

5. CONCLUSION

The memory issues of one-level and multi-level MCTF are explored in detail. The m-DCF scheme is proposed to reduce memory bandwidth considering fractional-pel MC. Many important system parameters are formulated, including computation complexity, memory bandwidth, external storage, and coding delay, which can be used as a reference for VLSI design. By case study, external memory bandwidth and storage size of multi-level MCTF are more than traditional MC prediction schemes. In the future work, we will use the open-loop property to reduce them. The coding delay needs to be reduced by modifying the coding structure, which is an important core experiment of MPEG-21 SVC.

6. REFERENCES

[1] D. Taubman, “Successive refinement of video: fundamental issues, past efforts and new directions,” in *International Symposium on Visual Communications and Image Processing*, 2003, pp. 791–805.

[2] ISO/IEC JTC1, “Call for proposals on scalable video coding technology,” ISO/IEC JTC1/WG11 Doc. N5958, Oct. 2003.

[3] ISO/IEC JTC1, “Scalable Video Model 3.0,” ISO/IEC JTC1/WG11 Doc. N6716, Oct. 2004.

[4] H. Schwarz, D. Marpe, and T. Wiegand, “MCTF and scalability extension of H.264/AVC,” in *Proc. Picture Coding Symposium*, 2004.

[5] C.-T. Huang, C.-Y. Chen, Y.-H. Chen, and L.-G. Chen, “Memory analysis of VLSI architecture for 5/3 and 1/3 motion-compensated temporal filtering,” in *Proc. ICASSP*, 2005.

[6] G. Pau, B. P.-Popescu, M. Schaar, and J. Vieron, “Delay-performance trade-offs in motion-compensated scalable subband video compression,” in *Advanced Concepts for Intelligent Vision Systems*, 2004.

[7] J.-C. Tuan, T.-S. Chang, and C.-W. Jen, “On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture,” *IEEE Trans. CSVT*, vol. 12, no. 1, pp. 61–72, Jan. 2002.