Direct macroblock coding for predictive (P) pictures in the H.264 standard

Alexis M. Tourapis*[a], Feng Wu[b], Shipeng Li[b]
[a]Thomson Corporate Research, 2 Independence Way, Princeton, NJ, USA 08550
[b]Microsoft Research Asia, 3F Sigma Center, 49 Zhichun Road, Beijing, 100080, China

ABSTRACT

In this paper we introduce a new Inter Macroblock type within the H.264 (or MPEG-4 AVC) video coding standard that can further improve coding efficiency by exploiting the temporal correlation of motion within a sequence. This leads to a reduction in the bits required for encoding motion information, while retaining or even improving quality under a Rate Distortion Optimization Framework. An extension of this concept within the skip macroblock type of the same standard is also presented. Simulation results show that the proposed semantic changes can lead to up to 7.6% average bitrate reduction or equivalently 0.39dB quality improvement over the current H.264 standard.

Keywords: Video Coding, H.264, Motion Compensation, Direct Prediction, Motion Prediction, Skip Macroblock.

1. INTRODUCTION

The motivation for increased coding efficiency in video coding standards has led to the adoption of more refined and complicated motion models and modes within the upcoming H.264 (or MPEG-4 AVC, JVT, H.26L) video coding standard[1]. These include Multiple-frame indexing of the motion vectors (MVs), increased sub-pixel accuracy, multi-referencing, and most importantly tree structured macroblock[2] and motion assignment, according to which different sub areas of a Macroblock are assigned to different motion information. A macroblock can essentially have up to 16 MVs since the tree macroblock structure enables the macroblock to be coded in 4 different modes and partitions of shape sizes equal to 16×16, 16×8, 8×16, and 8×8, while when in the 8×8 partition mode, each 8×8 partition can be further split into 8×8, 8×4, 4×8, and 4×4 blocks (Fig. 1). The standard also benefited from the definition and usage of a SKIP macroblock mode in Predictive (P) pictures, according to which, originally, a macroblock could be signaled as having zero motion and did not require the transmission of any residual information. Considering the high probability of occurrence of this mode, especially on stationary or quasi-stationary sequences, additional methods such as Run Length Coding were also introduced to further improve efficiency. Unfortunately, not all sequences are stationary, while the benefit from the above Inter methods reduced considerably at lower bitrates when considering the significant amount of bits that are required to encode the more precise mode and motion information.

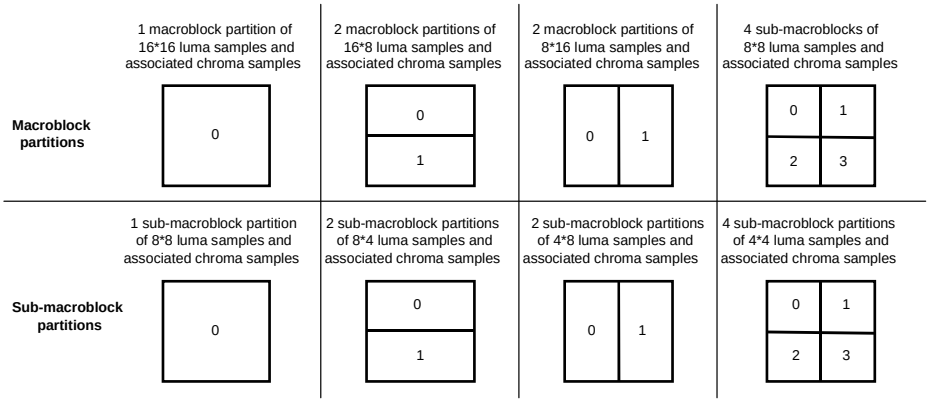


Fig. 1. Macroblock and sub-macroblock partitions as defined in H.264.

* alexismt@ieee.org; phone +1-(609)-987-7329; fax +1-(609)-987-7299

This problem becomes even more significant in the case of Bi-Predictive (B) pictures where a macroblock may be predicted using two motion vectors pointing to different pictures that are assigned to two different lists (list0 and list1). This essentially means that an even larger percentage of bits is required for the encoding of motion vectors and modes. Similar to SKIP, this problem was partially solved by the introduction of the Temporal Direct Prediction macroblock mode where, instead of encoding the actual motion information, by making the assumption that an object is moving with constant speed, both list motion vectors are derived directly from the motion vectors used in the co-located macroblock of the first list1 reference picture (Fig. 2).

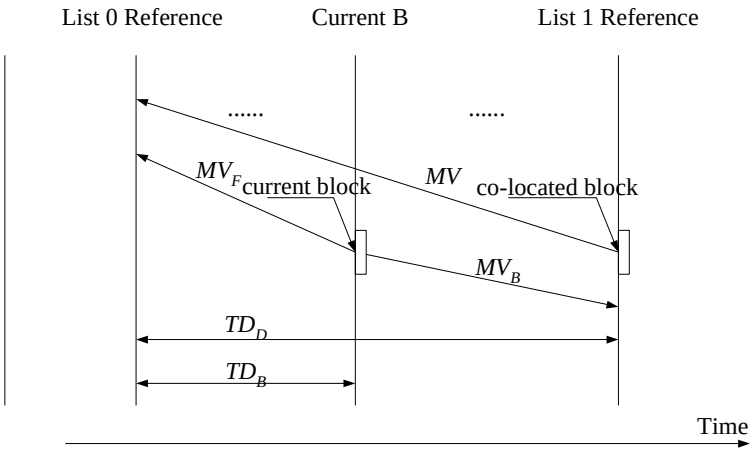


Fig. 2. Direct Mode in B picture coding.

Some other technologies previously proposed[3,4] tried to further improve efficiency and reduce the bits required by motion information, by introducing Global Motion Compensation (GMC) concepts within the standard. Unfortunately, these provided relatively small improvement in encoding performance while, at the same time, introduced a considerable increase in complexity in both the encoder and decoder. On the other hand, a much simpler technique, which only required a simple modification on the semantics of SKIP mode was introduced to the H.264 standard by Lainema[5], and could in general terms achieve much better performance than Global Motion Compensation methods. Considering that a macroblock has a much higher probability to have similar motion as its neighbors, SKIP mode was modified to consider a MV which was the median value of the MVs from the three adjacent, on the left, top, and top-right, blocks to the current macroblock, instead of zero (Fig. 3). The process is essentially very similar to the generation of the motion vector predictor used for coding the actual motion vectors, if such are available. However, an additional concept was introduced where if the macroblock was on the first row or column, or either the left or top adjacent macroblock MVs were equal to zero, the zero MV was still used. This concept is called zero partitioning of the SKIP mode.

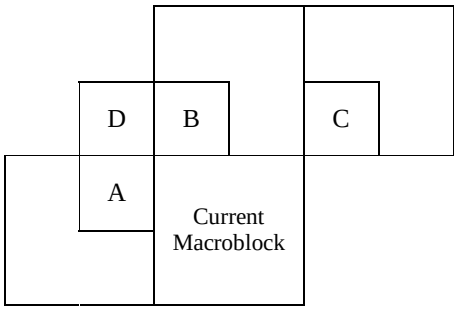


Fig. 3. Spatial predictors used for Motion Vector Prediction and the generation of the SKIP mode MV parameters.

In a sense this new MotionCopy SKIP mode also exploited spatial correlation to achieve better coding efficiency. MotionCopy allowed for up to 10% bitrate reduction or 2.2% bitrate reduction on average versus zero SKIP. A very similar concept was also introduced within B pictures[6], where unlike the temporal direct mode, direct mode parameters were generated through the usage of spatial predictors as in SKIP mode. On the other hand a different zero partitioning method was considered where temporal information was used instead. This was named as the Spatial Direct mode, and selection between temporal and spatial direct modes was performed at the slice level of the H.264 standard.

In this paper we introduce an additional method to further improve coding efficiency by extending the concept of the Direct Temporal Macroblock Mode to P pictures and by introducing a new mode named as the DirectP mode. Unlike MotionCopy that benefits from exploiting spatial correlation, our new mode essentially exploits temporal correlation that may still exist between macroblocks at temporally adjacent pictures. In addition, by also modifying the zero partitioning semantics of the MotionCopy mode to also consider temporal correlation similar to Spatial Direct mode in B pictures, coding performance is further improved.

In Section 2 we will first introduce our new DirectP macroblock mode. The modifications to MotionCopy and further extensions of the DirectP macroblock mode will be given in Section 3, followed by simulation results and our conclusion.

2. DIRECT PREDICTION

As previously discussed, Temporal Direct Prediction in B pictures can considerably improve coding efficiency. For this mode, the direct list0 and list1 MVs for the current macroblock are calculated as the temporally interpolated MV values of a corresponding co-located macroblock in the first list1 reference picture, while its associated reference pictures are the first list1 picture and the picture pointed to by the MV of the co-located block , for the direct list1 and list0 MVs respectively. The direct MVs are calculated according to the following equations:

$$Z = (TD_B \times 256)/TD_D \qquad \begin{aligned} MV_F &= (Z \times MV + 128) >> 8 \\ MV_B &= MV_F - MV \end{aligned} \tag{1}$$

where TD_B and TD_D are the temporal distance between the current B picture and its list0 reference and the temporal distance between the list1 and list0 references respectively. MV is the motion vector of the co-located block, and MV_F and MV_B are the final interpolated list0 and list1 motion vectors for the direct mode. Considering that the co-located macroblock can have up to 16 MVs, this mode can efficiently represent very complicated motion. It would be highly desirable to have a similar property within P pictures as well.

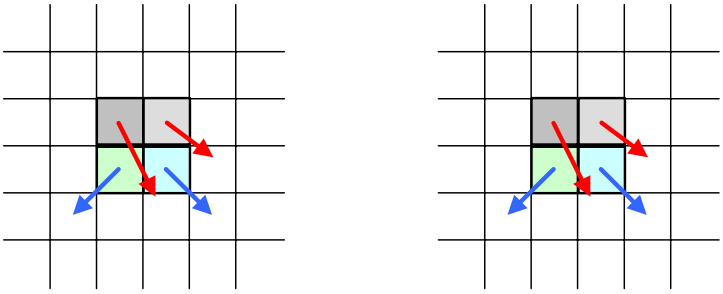


Fig. 4. Direct P Prediction. Collocated MBs have identical motion information.

It is rather obvious that the assumption on constant speed of an object that was used for temporal direct prediction in B pictures can also be used in P pictures as well. In particular, we may reuse the motion vectors of the co-located macroblock of the first list0 reference picture for P pictures as well. Here we can either choose to extrapolate motion

vectors while using the same references pointed to by the co-located, or simply copy the motion information and reference indices (Fig. 4), assuming that distance between references remains unchanged for each picture (Fig. 5). If distance changes, even though not necessary for such a scheme, a more accurate motion vector can be easily computed using the following equation:

$$Z = (TD_1 \times 256) / TD_2 \qquad MV_P = (Z \times MV + 128) >> 8 \tag{2}$$

where TD_1 and TD_2 are the temporal distance between the current picture and the reference of the current block, and the temporal distance between the first reference and the reference of its co-located. Although it is possible to select any reference for the current block, the most appropriate solution would be to select the closest reference since that would most likely have the highest correlation with the current block.

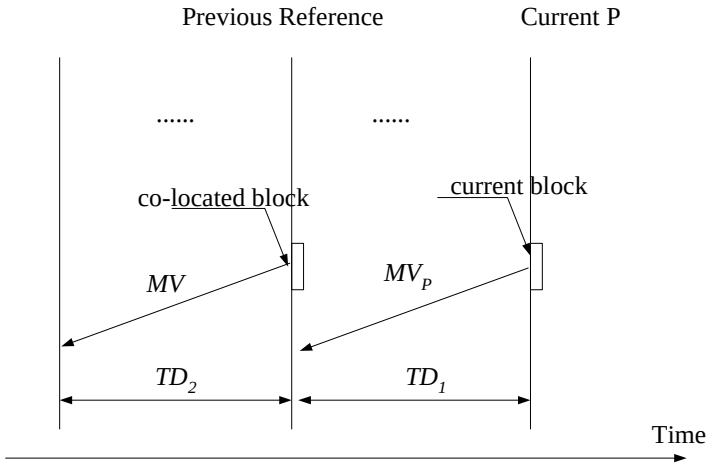


Fig. 5. Direct Mode in P picture coding.

The above concept can be easily introduced in the H.264 standard as an additional inter mode (DirectP) that follows in priority the 16×16 Macroblock type (Table 1). Although we may select not to transmit residual information with this mode similar to MotionCopy, we have found that transmitting the residual is more effective.

Table 1. Proposed Inter macroblock type ordering.

INTER MODES		***Description***
MotionCopy	0	MotionCopy Skip Mode
16x16	1	One 16×16 block
DIRECTP	*2*	*Direct Prediction mode*
16x8	3	Two 16×8 blocks
8x16	4	Two 8×16 blocks
8x8	5	Four 8×8 blocks using tree structure

A special consideration needs to be made for cases when the co-located macroblock is intra, or if the co-located macroblock points to reference pictures other than zero. For both cases several alternative solutions can be used. For example, for co-located intra we may use the first list0 reference picture and consider the block as having either zero, or the same median motion vector that is used for the MotionCopy mode, or for coding the 16×16 Inter mode MVs. Unlike MotionCopy, which might also be zero due to zero partitioning, considering that the DirectP mode allows the encoding of residual information, this consideration can provide an alternative prediction that can lead within a RDO framework to additional performance benefits. For the second case, although we may prefer to leave the motion information and references unchanged, we can again use the Median MV as an alternative, or even scale the MV to the

zero reference. The later makes more sense considering the higher correlation of this reference compared to other, more distant ones. Finally, this mode can also be extended to B pictures, as an additional single prediction direct mode, which on the other hand could also be combined with actual motion information for the secondary prediction. The benefit in such a case could be even more significant considering the much higher impact and importance of motion information within B pictures. On the other hand, a possible drawback of this mode is the additional memory required in both encoder and decoder to store all motion vectors and reference indices for a picture. Considering though that this information is already required and stored for the temporal direct mode, it is obvious that the increase in complexity can be completely ignored.

3. SPATIO-TEMPORAL MOTIONCOPY AND DIRECT PREDICTION EXTENSIONS

It would be highly desirable for the MotionCopy Skip mode to be able to also exploit temporal correlation apart from spatial, if it could lead to further increase in performance, and considering that it is the most efficient P picture macroblock mode within H.264. Unfortunately, we have found that replacing its motion vector derivation completely using a method similar to DirectP could potentially impair performance, especially when taking in consideration that spatial correlation is more dominant than temporal. Instead, we have found that a combination of both spatial and temporal correlation, as was done for the Spatial Direct mode, is more suitable and can achieve better performance. In particular, we observe that when a block is stationary, that is has zero motion vectors and reference index, it is also highly likely that its co-located in an adjacent picture would also be stationary. This condition also appears to be much stronger than the current zero partitioning rules used for the MotionCopy MV derivation.

Using the above observation, we completely replace the zero partitioning rules with the consideration of zero MVs within the co-located partition, while leaving the rest of the process completely unchanged. This enables the simultaneous exploitation of both spatial and temporal correlation and thus improving performance. We should point out that this means that a MotionCopy macroblock may have two motion vectors since it's co-located may have partitions with both zero and non-zero MVs. More specifically the derivation of the motion parameters of a MotionCopy macroblock is performed as follows:

Step 1. Calculate the 16×16 motion vector predictor $PMV_{16\times16}$ using the first list0 reference for the current macroblock.
Step 2. For every 4×4 block sub-partition within the current macroblock, examine whether it's associated co-located 4×4 block sub-partition within the first list0 reference has zero MVs and reference. If yes go to Step 4, otherwise go to Step 3.
Step 3. Use the $PMV_{16\times16}$ and the zero list0 reference as the motion parameters for the current 4×4 block sub-partition and exit.
Step 4. Use the zero MV and list0 reference as the motion parameters for the current 4×4 block sub-partition and exit.

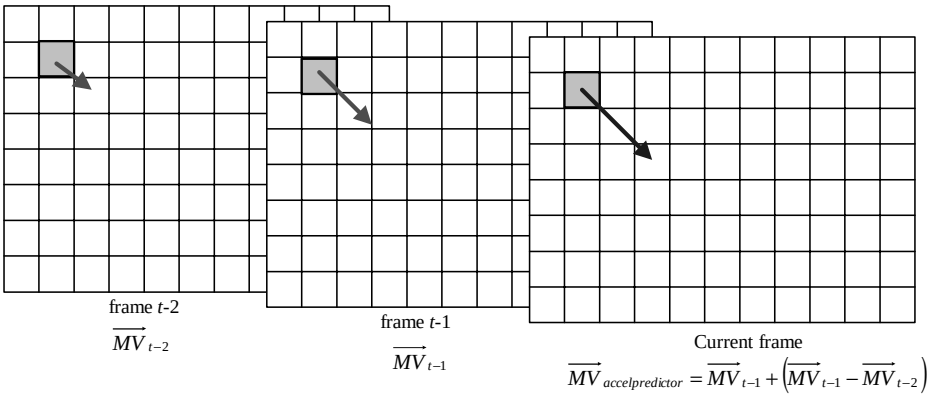


Fig. 6. Usage of acceleration information for Direct Prediction.

Apart from the modification to MotionCopy, we can also improve the performance of Direct Prediction by considering additional concepts such as acceleration, or by introducing multi-hypothesis prediction (combination of temporal and spatial MVs). Instead of copying, for example, the motion vectors of the co-located block from the first reference, we use motion vectors from two different reference pictures and consider also an acceleration factor between the two references (Fig. 6) for generating the motion parameters. MVs could for example be computed as:

$$Z_1 = (TD_1 \times 256) / TD_2$$
$$Z_2 = (TD_1 \times 256) / TD_3 \qquad (3)$$
$$MV_P = (2 \times Z_1 \times MV_{t1} - Z_2 \times MV_{t2} + 128) >> 8$$

where now MV_{t1} and MV_{t2} correspond to the motion parameters from the co-located blocks within the two reference pictures, and TD_3 is the distance between the second reference and the associated reference of the associated co-located block. Although this method could provide us with some additional benefit, it nevertheless increases complexity further, while it also requires more memory for the storage of the additional motion parameters needed for such computation.

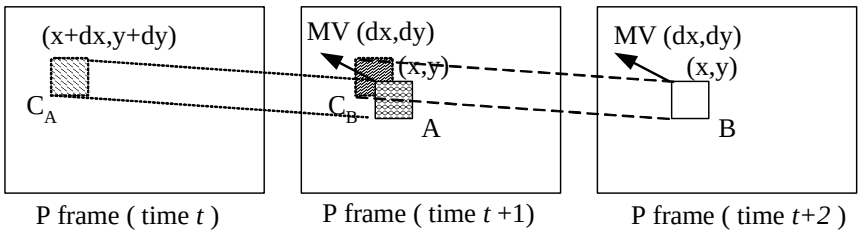


Fig. 7. Direct Motion Projection in P pictures

We also observe that the current DirectP mode is mainly concerned with projecting motion vectors from one block in one picture to its co-located in the next picture (motion projection), as can also be seem in Fig. 7. However, it was previously observed [7], that it is more precise, instead of projecting motion vectors (Fig. 8) to project the pixels of a moving block to a new position using their associated motion vectors (pixel projection). Positions that cannot be predicted using pixel projection could be predicted using either motion projection, zero motion, or by considering the motion of adjacent available pixels. This process requires the generation of an additional reference picture on both encoder and decoder, and might be considered too complicated for some implementations. Nevertheless, this concept, or even a combination of both pixel and motion projection could lead to even further benefits, which will not be though investigated further in this paper.

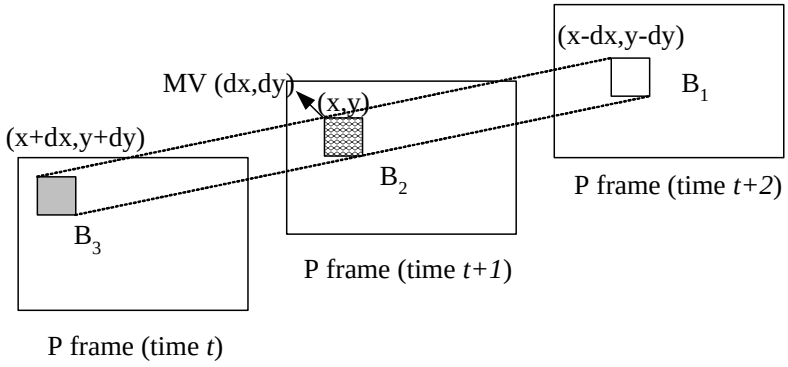


Fig. 8. Direct Pixel Projection in P pictures.

4. SIMULATION RESULTS

The DirectP mode and the semantic modification to MotionCopy were introduced within version 4.3a of the H.264 reference software[8]. For our simulations we have selected 5 sequences, namely QCIF (176×144) resolution sequences *Container* and *News* coded at 10fps, and CIF (352×288) resolution sequences *Mobile*, *Bus*, and *Flowergarden* at 30fps. The CAVLC entropy coder was used for all our tests, with quantizer QP values of 28, 32, 36, and 40, a search range of ±32, and 5 references. Rate Distortion Optimization was enabled in our simulations. To simplify our comparisons, we have used the method use for evaluating all JVT proposals[9]. This method enables the computation of the average PSNR gain (dPSNR), or equivalently the average bitrate reduction (Δbitrate), for the proposed method through the consideration and integration of the difference between the original and proposed methods' rate distortion curves given a set of four points generated at different quantization values..

Our complete results on Δbitrate and dPSNR are shown in Table 2. We immediately observe the considerable bitrate reduction for sequences Container (7.59%) and Mobile (6.23%), which are characterized by relatively constant and smooth motion. Sequences News, Bus, and Flowergarden have smaller, although not negligible, benefits (1.90%, 3.03%, and 3.10% respectively). The Rate Distortion curves for Container and Mobile are shown in Fig. 9. Furthermore, we observe that the benefits of the proposed scheme tend to increase at lower bitrates (approximately 0.5dB gain at 200kbps), which is to be expected considering that motion information can take a larger percentage of the coded information at low bitrates, while our proposed method allows for better representation of motion without having to spend as many bits in coding motion.

Table 2. Performance Evaluation of the Proposed Scheme.

Sequences	***Container***	***News***	***Mobile***	***Bus***	***Flower***
Δbitrate %	-7.59	-1.90	-6.23	-3.03	-3.10
δPSNR	0.391	0.115	0.300	0.157	0.161

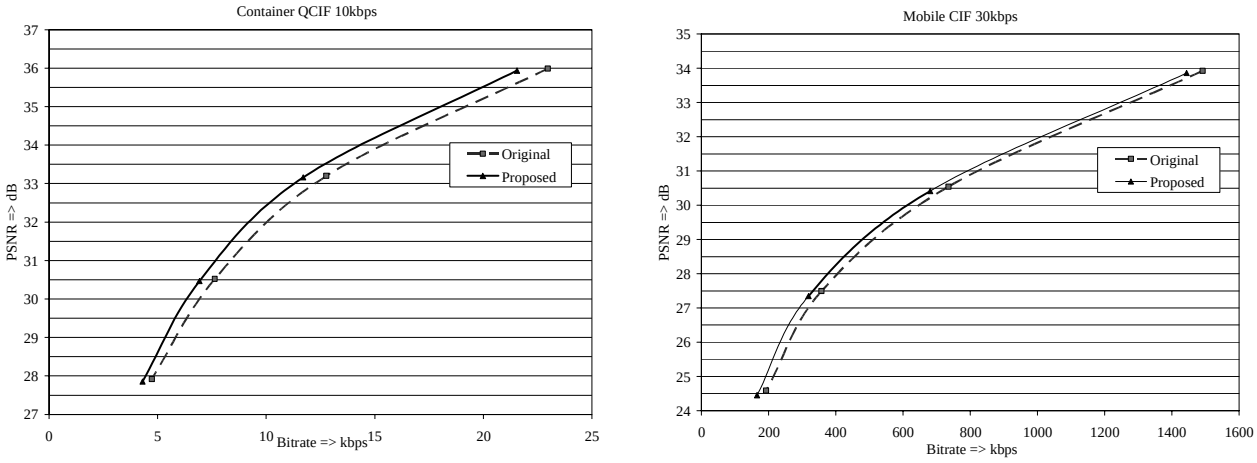
It is probably quite obvious from the algorithmic description of the DirectP mode and Direct Prediction in general, that the complexity introduced is rather small and mainly involves an increase of the memory requirements of the encoder and decoder. In particular, if no acceleration is used, it would be necessary apart from storing all reference pictures, to also store the motion vectors and associated reference indices of the first list0 reference picture. In the Direct Motion Projection case, and if we assume that no scaling of the co-located motion vectors is applied, the encoder only needs to examine one additional macroblock mode within the RDO process for which the motion vectors are taken immediately from their co-located, while at the decoder if a DirectP mode is encountered motion information is again used in similar fashion. On the other hand, GMC methods require apart from the computation of a certain set of GMC parameters, the generation of a usually warped reference image, which is obviously much more complicated than simple memory store and copy operations. Even if Direct Pixel Projection, acceleration, and the precise scaling of motion are considered, the increase in complexity would still obviously be considerably lower than that of warping, since these mainly involve simple computations. Furthermore, considering the multiple picture reference nature of H.264, and that other modes, such as temporal direct, already require the same motion information for the computation of their own parameters, we may deduce that the complexity required by such a method is relatively negligible. Finally the complexity introduced by the the modified MotionCopy Skip mode is negligible, while the concept unifies further the semantics of Skip mode in P pictures and Spatial Direct Mode in B pictures, which could be valuable in certain implementations.

5. CONCLUSION

In this paper we have presented a new Inter Macroblock type which could be incorporated into the H.264 or other video coding standards and architectures, and which tries to exploit the temporal correlations of motion vectors. A semantic change was also introduced to the MotionCopy Skip mode of H.264 that jointly considers temporal and spatial correlation. Simulation results demonstrate that our proposed modified codec can achieve much better coding efficiency especially at lower bitrates compared to the existing H.264 standard, with relatively little increase in complexity.

6. REFERENCES

1. Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, "Joint Video Specification (ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC) - Joint Committee Draft," document JVT-E022d3.doc, Sep'02.
2. Heiko Schwarz and Thomas Wiegand, "Tree-structured macroblock partition," document VCEG -O17, 15th VCEG meeting, Pattaya, Dec'01.
3. Hideaki Kimata, "GMVC and GMC Switched by MV," document JVT-B046, 2nd JVT meeting, Geneva, Jan'02.
4. Shijun Sun and Shawmin Lei, "Global Motion Vector Coding (GMVC), " document JVT-B019, 2nd JVT Meeting, Geneva, Jan'02.
5. Jani Lainema and Marta Karczewicz, "Skip mode motion compensation," document JVT-C027, 3rd JVT Meeting, Fairfax, May'02.
6. A. M. Tourapis, F. Wu, S. Li, "Direct mode coding for bi-predictive pictures in the JVT standard," in proceedings of the 2003 IEEE International Symposium on Circuits and Systems (ISCAS'03), pp.700-703, Bangkok, Thailand, May'03.
7. A. M. Tourapis, H. Y. Cheong, M. L. Liou, and O. C. Au, "Temporal Interpolation of Video Sequences Using Zonal Based Algorithms," in proceedings of the 2001 IEEE International Conference on Image Processing (ICIP'01), WP8-5252, Thessaloniki, Greece, Oct'01.
8. JVT Reference Software unofficial version 4.3a, http://bs.hhi.de/~suehring/tml/download/Unofficial/
9. G. Bjontegaard, "Calculation of average PSNR differences between RD-Curves," document VCEG-M33, 13th VCEG meeting, Austin TX, Mar'01



(a) (b)

Fig. 9. RD performance plot for sequences a) Container at 10fps and b) Mobile at 30fps.