Enhanced Inter Prediction via Shifting Transformation in the H.264/AVC

IZQUIERDO, E; Blasi, SG; Peixoto, E

“The final publication is available at
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6279460”

Enhanced Inter Prediction via Shift Transformation in the H.264/AVC

Saverio G. Blasi, *Student Member, IEEE*, Eduardo Peixoto, *Student Member, IEEE*
and Ebroul Izquierdo, *Senior Member, IEEE*

***Abstract*—Inter-prediction based on block-based motion estimation is used in most video codecs. The closer the prediction is to the target block, the lower is the residual and more efficient compression can be achieved. In this paper a new technique called Enhanced Inter-Prediction (EIP) is proposed to improve on-the-fly the prediction candidates using an additional transformation acting while performing motion estimation. A parametric transformation acts within the coding loop of each block to modify the prediction for each motion vector candidate. The EIP is validated in the particular case of a single-parameter shifting transformation. This paper presents an efficient algorithm to compute the best shift for each prediction candidate, and a model to select the optimal prediction based on minimum cost integrating the approach with existing rate-distortion optimization techniques in the H.264/AVC video codec. Results show significant improvements with an average of 6% bit-rate reduction compared to the original H.264/AVC.**

***Index Terms*—Inter prediction, video coding, H.264/AVC**

I. INTRODUCTION

INTER-prediction based on multiple reference frames and variable block size Motion Estimation (ME) is used in the state-of-the-art H.264/AVC standard [1] in order to achieve compression exploiting temporal redundancy. By means of a set of Motion Vectors (MV) the encoder computes a prediction and calculates the residual difference with the original target. The residual frame is encoded instead of the original target, resulting in a significantly smaller bit-stream. The parameters needed to compute the inter-frame prediction need also to be encoded in the bit-stream, included in what is generally referred to as motion information.

Several techniques have been proposed in the past to improve this scheme based on frame transformation. In order to enhance the motion compensated prediction (MC), sub-pel ME based on reference frame interpolation [2] has been successfully implemented in the context of inter-prediction. Despite requiring larger motion information to transmit fractional MV components, sub-pel techniques achieve higher compressions due to smaller residuals that need less bits to be encoded. Sub-pel prediction is included in several standards, including the H.264/AVC [3], which uses up to quarter-pel interpolation.

A technique that aims at improving inter-prediction by transforming the reference frame is global ME [4]. According to this method, a new reference frame is created using the previous encoded frames and the motion of these frames. This new frame is then added to the list of possible reference frames for the current frame. A similar approach is used in the Advanced Simple Profile of the MPEG-4 Part 2 [5]. Also, a work on luminance transform of the reference was proposed [6] as an alternative to ME, however, the method is exclusively applied to blocks where conventional motion-based inter-prediction is unsatisfactory. Pixel transformation using multiplying coefficients was also proposed [7] to address the specific problem of static scene changes in particular video sequences.

Weighted Prediction (WP) has been extensively investigated to improve MC prediction, officially included as part of the H.264. The standard allows two WP modes: implicit (used mainly to improve bi-prediction) and explicit (used mainly to compensate fade-to-black transitions) [8]. When using WP, the reference pictures used for ME for the current frame are transformed using a weighting factor and additive offset. In the explicit mode these parameters are computed at the encoder side for each reference frame using appropriate functions of this reference and current frame (mainly least mean square is used and implemented in H.264 JM reference software [9]). The parameters are then transmitted in the slice header and used at the decoder side on the references. In implicit mode, the parameters are not transmitted but computed at both encoder and decoder side, usually as a function of the temporal distance between current frame and each reference. Finally, when performing ME the transformed references are used instead of the original ones.

In the emerging HEVC standard [10], a filter is considered that also makes use of frame transformation, called Sample Adaptive Offset (SAO) [11]. The SAO parameters are found after the whole frame is encoded (i.e. after de-quantization and de-blocking filter) unlike previous techniques that use reference frames prior to quantization. Also, SAO has the different objective of compensating the error between original and reconstructed frames, hence only increasing the PSNR but no effects on the residuals or on inter-prediction. In particular, the encoder classifies the pixels in the reconstruction frame into different categories according to a set of rules, and thus extracts an optimal offset from a set of candidates in a look-up table, in a per-frame basis for each category. At the decoder, the relevant offset is applied to all the pixels in a frame belonging to the same category. The number of categories and therefore the amount of side information transmitted is very small.

These techniques aim at improving compression via transformations that operate at a frame level, either after MC

The authors are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, Mile End Road, London, E1 4NS, UK e-mail: sgblasi@eecs.qmul.ac.uk, eduardopeixoto@ieee.org, ebroul.izquierdo@eecs.qmul.ac.uk.

Manuscript received ?, ?; revised ?, ?.

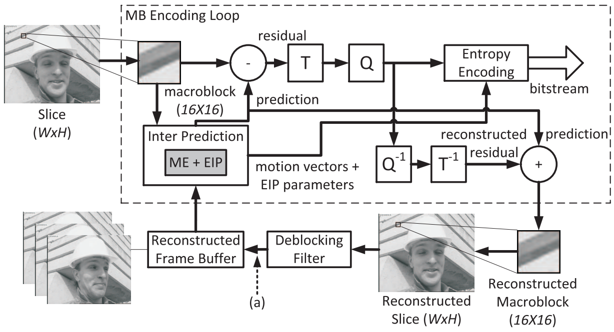


Fig. 1. Simplified diagram of a H.264/AVC encoder with EIP.

(as the WP) or after reconstruction (as the SAO). In this paper we instead investigate transformations that act locally, at a prediction level. The proposed approach is based on an additional module (called enhanced inter-predictor EIP) operating on-the-fly within the ME block in the MB encoding loop. The EIP consists of a parametric transformation acting on each prediction candidate at a given displacement for a tested MV; as the transformation directly changes the pixels in the current candidate, its associated prediction cost is also different. Eventually, the aim of the modified encoder with EIP is to find on-the-fly the optimal pair comprising both transformation parameters and optimal MV, that concurrently minimise the overall prediction cost. A simplified diagram of the modified encoder is shown in Fig. 1. While techniques such as global ME or explicit WP operate outside the MB coding loop as in point (a), and the SAO operates before storing the pictures in the buffer and displaying as in point (b), the EIP acts directly inside the loop among predictions during ME. At the encoder side, the EIP computes the optimal parameters for each MV candidate, and the encoder selects the best enhanced prediction. The optimal EIP parameters required to obtain the enhanced prediction are then transmitted in the bit-stream along with the MV components, and finally decoded to compute such prediction at the decoder side.

While this concept is theoretically feasible using a variety of transforming functions, in fact the EIP can only be efficient if the parameter computation adds little complexity and its transmission requires a relatively small amount of bits. A successful instantiation of the EIP is therefore detailed in this paper using a simple transformation, which we refer to as shifting transformation (ST). As a proof of concept, the approach is integrated in H.264/AVC using JM reference software [9], but the concept could be integrated in any codec following the DPCM/DCT model.

II. ENHANCED INTER-PREDICTION USING ON-THE-FLY PARAMETRIC TRANSFORMATIONS

Conventional block-matching ME algorithms imply the testing of a succession of prediction candidates (i.e. blocks in a reference frame at given displacements). The sequence and number of candidates vary depending on the ME algorithm used. The optimal MV is selected corresponding to the best candidate in terms of a specific error metric or cost (e.g. the sum of absolute differences, SAD). Formally, given a target block $\mathbf{T}$, the encoder outputs the prediction $\mathbf{P}_0$ such that $cost(\mathbf{P}_0, \mathbf{T})$ is minimum among the considered candidates $\mathbf{P}$.

Consider a transformation acting on a candidate $\mathbf{P}$ as in:

$$\mathbf{EIP} = \Theta\left(\mathbf{P}, x_1, x_2, ...\right) \tag{1}$$

Consider an array of values $(x_1, x_2, ...)_1$ such that the $\mathbf{EIP}_1$, computed using these values with Eq. 1 on the candidate $\mathbf{P}_1$ is such that $cost\left(\mathbf{EIP}_1, \mathbf{T}\right) \leq cost\left(\mathbf{P}_0, \mathbf{T}\right)$. The decoder can use $\mathbf{EIP}_1$ instead of $\mathbf{P}_0$ to produce smaller residuals, given that it is provided with the array $(x_1, x_2, ...)_1$. If the rate required to encode the array is compensated by sufficient gains due to the smaller residual, the method achieves lower bit-rates at better reconstruction qualities.

Clearly the EIP is only implementable if the number of parameters in Eq. 1 is small and their computation is efficient. Hence, the approach was developed here using a simple transformation (shifting transformation, ST), defined by a single additive parameter acting on a prediction candidate as in:

$$\mathbf{EIP_{ST}} = \Theta\left(\mathbf{P}, s\right) = \mathbf{P} + s \tag{2}$$

Notice again that, if the idea of an additive offset is not new in video coding and the proposed transformation shares similarities with other approaches, the EIP is based on a novel concept and its implementation and results differs greatly from previously introduced techniques. Methods such as SAO or WP contribute with refinements that are found externally to the MB coding loop. In the case of WP, the WP parameters are computed prior to encoding a frame based on global features of current frame and each reference; once these parameters are selected according to the mode and method being used, and after the references are correspondingly transformed, such references are used for ME throughout the whole frame. The EIP operates instead within the MB coding loop on each inter-prediction candidate, and as such it has a very different impact on the outcomes of inter-prediction. The computation of the optimal shift s is performed on-the-fly only depending on target block and candidate prediction block pointed by the currently tested MV. As such the ST parameter can also be optimised to trade-off between rate and distortion similarly to the selection of the best MV among the successively tested candidates. Eventually, the amount of side information transmitted is largely different, as in the EIP with ST one shift has to be transmitted for each transmitted MV. Such differences are crucial in allowing the approach to work in a wide combination of conditions. For instance results show that while WP in explicit mode works effectively only under particular scene conditions in certain frames, the EIP with ST consistently provides enhancements on a frame-by-frame basis even for long sequences.

We propose in this paper an efficient technique to compute the ST parameter on-the-fly for a given prediction candidate, and present a novel framework to embed the EIP in typical temporal prediction schemes supported by rate-distortion optimization. The EIP using ST can be implemented and optimised to significantly improve inter-prediction in a large variety of cases bringing significant gains to the coding performance, as shown in the following sections.

III. OPTIMAL SHIFT CALCULATION

The derivation of a method to compute the optimal ST parameter and associated distortion are shown here. The calculation is derived using the Sum of Absolute Differences (SAD) as the distortion measure (other measures could be used, but the algorithm would need some adaptation). The method outputs, in a single step, the optimal shift and the corresponding SAD offset for that shift.

Consider the difference block between a prediction and the current target block, and arrange its values in a vector $\mathbf{D}$. We refer to each element in $\mathbf{D}$ as $d(i) = p(i) - t(i)$, $i = 0, 1, ..., (B-1)$ where $B = M \times N$, M and N are the block height and width, and $p(i), t(i)$ are the elements in prediction and target block respectively. Consider now with no loss of generality that the elements in $\mathbf{D}$ are rearranged in increasing order, i.e. $d(i) \leq d(j) \forall i \leq j$. Define as $SAD(s) = SAD(\mathbf{EIP}, \mathbf{T}) = SAD(\mathbf{P}+s, \mathbf{T})$ the SAD of the enhanced prediction and target. In particular for a shift $s = 0$ we have:

$$SAD\left(0\right) = \sum_{i=0}^{B-1} \left|d\left(i\right)\right| = \sum_{i=0}^{B-1} \left|p\left(i\right) - t\left(i\right)\right|$$

Notice that due to the commutative property, the rearrangement does not affect the SAD. Denote now with $N_+(0)$, $N_0(0)$ and $N_-(0)$ the number of positive, zero and negative elements in $\mathbf{D}$ respectively, which we refer collectively as the sign ratio. According to the rearrangement we have $d(i) < 0$ for $i = 0, ..., (N_-(0) - 1)$ and correspondingly:

$$SAD\left(0\right) = -\sum_{i=0}^{N_-(0)-1} \left(d\left(i\right)\right) + \sum_{i=N_0(0)+N_+(0)}^{B-1} \left(d\left(i\right)\right)$$

If we modify the current prediction with a unitary positive shift, we obtain a new difference vector $\mathbf{D_{+1}}$ with elements $d_{+1}(i) = (p(i) + 1) - t(i)$. Define with $SAD(+1)$ the SAD of the modified prediction and target. Then:

$$\begin{aligned} SAD\left(+1\right) &= \sum_{i=0}^{B-1} \left|d\left(i\right) + 1\right| \\ &= -\sum_{i=0}^{N_-(0)-1} \left(d\left(i\right)+1\right) + \sum_{i=N_0(0)}^{B-1} \left(d\left(i\right)+1\right) \\ &= SAD\left(0\right) - N_-\left(0\right) + N_0\left(0\right) + N_+\left(0\right) \end{aligned}$$

If the additional factor $(-N_-(0) + N_0(0) + N_+(0))$ is negative, we are enhancing the prediction ($SAD(+1) < SAD(0)$). This happens if and only if the number of negative elements in $\mathbf{D}$ is larger than the remaining number of zero and positive elements, or if:

$$N_-\left(0\right) > N_0\left(0\right) + N_+\left(0\right)$$

which we refer to as *Condition 1*. The additional factor $(-N_-(0) + N_0(0) + N_+(0))$ is in fact equal to the offset between the non-shifted SAD and the enhanced SAD.

Consider now that *Condition 1* is satisfied for the current prediction-target couple. Remind that the elements in $\mathbf{D}$ are sorted in increasing order. Denote with $k = d(N_-(0) - 1)$ the last negative element in $\mathbf{D}$ (i.e., the maximum negative element). Assume, as an example, that k is unique. If we apply a positive shift s to the prediction, such that $s < |k|$, *Condition 1* is still met because $N_-(+s) > N_0(+s) + N_+(+s)$. The SAD offset is equal to $\Delta(SAD) = s \cdot (-N_-(0) + N_0(0) + N_+(0))$.

If we apply a shift equal to $s = |k| = -d(N_-(0) - 1)$, this element would become a zero in the shifted difference vector, resulting in $N_0(+s) = 1$, $N_+(+s) = N_0(0) + N_+(0)$ and $N_-(+s) = N_-(0) - 1$. In order for *Condition 1* to hold for the shifted vector $\mathbf{D_s}$, the sign ratio for the original vector $\mathbf{D}$ would need to satisfy $N_-(0) > N_0(0) + N_+(0) + 2$.

If the aforementioned condition is not satisfied, a larger shift increases the SAD. The optimal shift is then: $-d(N_-(0) - 1)$, with an SAD offset of $(d(N_-(0) - 1) \cdot (-N_-(0) + N_0(0) + N_+(0)))$. If the condition is satisfied, we still decrease the SAD by further increasing the shift. This happens at each successive positive unitary shift, until *Condition 1* is no longer met.

By iterating and generalising this idea, we see that if *Condition 1* is initially met, we can successfully decrease the SAD using a shift as large as:

$$s = -d\left(N_-\left(0\right) - n_{max}\right) \tag{3}$$

where n_{max} is the maximum integer number n such that $N_-(0) > N_0(0) + N_+(0) + 2 \cdot n$. Thus:

$$n_{max} = floor\left(\frac{N_-\left(0\right) - N_0\left(0\right) - N_+\left(0\right)}{2}\right) \tag{4}$$

The corresponding SAD offset as a result of the ST is:

$$\begin{aligned} \Delta\left(SAD\right) &= SAD\left(0\right) - SAD\left(-d\left(N_-\left(0\right) - n_{max}\right)\right) \\ &= \sum_{n=0}^{n_{max}-1} -\left(N_0\left(0\right) + N_+\left(0\right) - N_-\left(0\right) + 2\cdot n\right) \cdot \\ &\qquad \left[d\left(N_-\left(0\right) - n - 1\right) - d\left(N_-\left(0\right) - n\right)\right] \end{aligned} \tag{5}$$

Using Equations 3, 4 and 5 we can compute an optimal positive shift in the case *Condition 1* is satisfied, with its corresponding SAD offset, in a single step from a given prediction and target blocks. The same method can be derived for a negative shift, with similar equations.

IV. FRAMEWORK IMPLEMENTATION IN THE H.264/AVC

The EIP in the particular case of ST is implemented in the H.264/AVC standard using the JM reference software 18.2 [9]. Intra or skipped blocks are left unchanged. H.264/AVC inter-prediction allows the 16×16 inter macroblock (MB) to be partitioned in smaller prediction units (sub-blocks of 16×8, 8×8 and down to 4×4). The optimal shift is computed for each MV candidate (using Eq. 3 and 4 or their negative equivalents) in each prediction unit: for instance two shifts are computed for the 16×8 mode for the top and bottom blocks respectively.

First, some tests were performed without considering rate-distortion (RD) issues or the cost of transmission for the shifts, in order to show and analyse the shift values distribution. As an example, Fig. 2 shows the histogram of the optimal shifts output by the encoder for the first 8 frames of the Foreman sequence (QP = 22). Similar results were obtained in all tests. The distribution is clearly centered in zero with a low deviation: even without considering RD cost, still around one fifth predictions are encoded with an optimal shift equal to zero. Also, large shifts are in general very rare with the vast majority being in absolute value smaller than 20 (even if some values are found as high as 256). In the light of these results the encoder was modified to output the optimal pair of MV and shift (MV, s) in a RD sense.

In particular, each time a shift $s_i \neq 0$ is found for a candidate MV_i, the enhanced SAD (found using Eq. 5) is used to compute the current RD cost as in:

$$J_i = SAD\left(s_i\right) + \lambda_{MV} R_{MV_i} + \lambda_S R_{S=s_i} \qquad (6)$$

The encoder keeps track of the best enhanced solution in the form of the pair (MV_{i1}, s_{i1}) that minimises this cost. Following again from the distribution in Fig. 2, it is obvious that the standard non-enhanced solution (i.e., with zero shift) has a relatively high probability of being selected, therefore the non-enhanced SAD is always computed for each candidate. In particular, the encoder keeps track of the best non-enhanced solution for the current block in the form of the pair $(MV_{i2}, 0)$ that minimises the cost computed again using Eq. 6. Notice that the transmission cost of the zero valued shift is considered in this case.

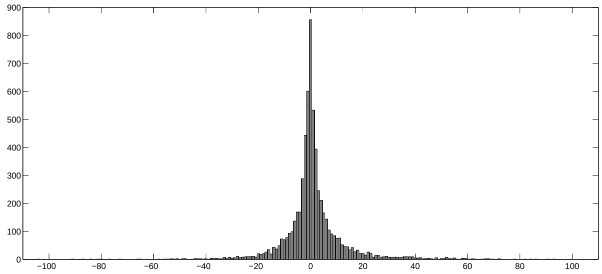


Fig. 2. Histogram of the optimal shift values for the first 8 frames of the Foreman sequence encoded with QP = 22.

After all the MV candidates are tested for a given prediction unit, the algorithm compares enhanced and non-enhanced solutions. While other techniques could be used for this comparison (for instance, the actual number of bits needed for coding could be used), in this paper the solution with minimum cost J_{i1} or J_{i2} was selected.

Following again from the histogram in Fig. 2, we assumed for the shift values a similar distribution as the MV difference (MVD) between MV and MV prediction (MVD values are in fact also centered in zero with a low variance). For this reason, the ST parameter is encoded in the modified encoder using the same VLC tables used for the MVD. These require few bits for small values and progressively more bits for higher, less probable values. For the same reasons the same Lagrangian multiplier λ used in the JM reference software for computing the MV cost is used to compute the shift cost (i.e. $\lambda_S = \lambda_{MV}$ in Eq. 6).

Finally the bit-stream was also modified to include the ST parameter. The shift is encoded immediately after the MVD components, using the same Exp-Golomb codes [3]. In this paper no prediction of the ST parameter is considered among blocks in the same frame, even if some correlation was found among neighbouring blocks (this is being considered for further developments). Finally, the shift parameter is limited to the range $-20 < s < 20$ in the current implementation of the modified encoder.

As the bit-stream is modified to accommodate the additional EIP parameters, a suitable decoder needs to be implemented accordingly. While the computation of the shift does add some complexity to the modified encoder (which is anyway limited by the efficiency of the algorithm in Section III), the alterations to the decoder (needed to decode the shift values and use these while computing the reconstructed frame) produce almost no impact on the decoder complexity. It is important to notice that the EIP drastically changes the outcomes of inter-frame prediction at all levels. Due to its impact on ME, the EIP leads to different MV than conventional inter-prediction. Due to different prediction costs, it has a relevant role on the selection of the best coding mode for an MB. Finally, due to different MV predictions, it might also change outcomes of the SKIP mode for subsequent MB.

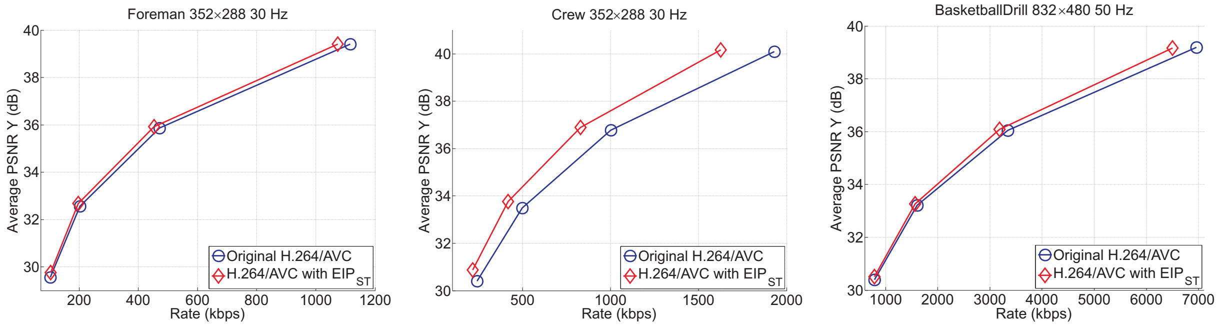
V. RESULTS

The approach was tested on several popular sequences at different resolutions. In all cases, the full length of the sequences is used (e.g. 300 frames are tested for sequences at 30 Hz). The coding configuration used is $IPPP$ with one reference frame, and the entropy coding used is the CAVLC. Four QPs are used: $22, 27, 32, 37$. Selected results are shown in Figure 3. Full results are shown in Table I in terms of the BD bit-rate (in percentage) [12], compared with WP in explicit mode using LMS (least mean square) method.

The proposed approach outperforms the original JM in all sequences but Mobile at 7.5 Hz, where it increases the bit-rate by an average of 0.6%. Visually, the approach does not change the subjective quality of the reconstructed video. For most sequences, the gain achieved is between 4 to 6% in terms of BD-rate, generally distributed towards high reductions in the bit-rate and unaffected reconstruction PSNRs, but considerable gains in PSNR were obtained in some cases (as shown on Fig. 3). Some sequences result in lower gains (2.3% gains for Party Scene, and 0.3% for Mobile 30 Hz). For two sequences, Crew and Waterfall, the approach shows a significantly large gain (up to 19% for Crew and 11% for Waterfall).

These results were compared with those obtained using WP in explicit mode, used again against original non-weighted JM. The method used to compute the WP parameters is LMS (as in the JM). Best results were obtained when chroma support was disabled and weighted references were used for ME in the encoder configuration. Results are shown in Table I.

The WP in explicit mode is clearly mostly ineffective when used on long sequences, providing unchanged or slightly worse rates on almost all the tests. To understand and clarify these results, we analysed some frame-by-frame examples, particularly those where the EIP with ST resulted instead in very large BD-rate gains. In the Crew sequence at 30 Hz with QP=32, the WP outputs impressive results on some of the first frames with up to 35% bit-rate reductions and some gains in PSNR. When tested on the first 4 frames of the



(a) (b) (c)

Fig. 3. Results for: (a) Foreman 352 × 288 30 Hz; (b) Crew 352 × 288 30 Hz; and (c) Basketball Drill 832 × 488 50 Hz.

TABLE I
BD RATE OF PROPOSED ST APPROACH AND EXPLICIT WP AGAINST CONVENTIONAL H.264/AVC.

Resolution	Sequence	FPS	BD-rate ST	BD-rate WP
352 × 288	Carphone	30	−6.0	0
	Foreman	30	−4.7	0
	Crew	30	−19.5	−0.1
	Mobile	30	−0.3	0
	Mother-Daughter	30	−4.0	0
	Bowing	30	−4.2	0
	Waterfall	30	−11.6	−0.1
	Carphone	7.5	−5.3	0
	Foreman	7.5	−4.6	+0.1
	Crew	7.5	−13.3	0
	Mobile	7.5	+0.6	0
	Mother-Daughter	7.5	−5.0	0
	Bowing	7.5	−10.1	+0.1
	Waterfall	7.5	−9.9	−0.2
832 × 480	PartyScene	50	−2.4	0
	BasketBall Drill	50	−4.8	0
	RaceHorses	30	−4.1	0

sequence, the WP is able to produce a 7.6% BD-rate gain against conventional non-weighted prediction. This is better than what is obtained with the proposed EIP with ST, only capable of producing a 4.5% gain on the same 4 frames. When testing larger amount of frames though, the performance of explicit WP is highly influenced by generally unaffected or even bad outputs in the vast majority of the frames where no brightness transitions or fades are present; in these cases the explicit mode actually deteriorates the encoder performance mostly due to the explicit transmission of the parameters. In the end, in average only 0.1% reduction in BD-rate is obtained for the whole 300 frame Crew sequence. Best results for the WP were obtained in the Waterfall sequence at 30 Hz, where the WP in the best coding configuration provided a 0.2% BD-rate reduction. Again this is a consequence of very large gains in some sparse frames and almost no effect in the remaining frames. The EIP with ST instead provided consistent results among all the frames in the tested sequences, providing some bit-rate reduction and/or quality improvement in the vast majority of the cases. Eventually, notice that the EIP approach has been also tested on top of the WP in explicit mode on some sequences. While this paper does not present extensive tests in these conditions, the modified encoder provided very similar results and performance as these obtained using EIP on top of non-weighted prediction.

VI. CONCLUSIONS

We propose an original inter-frame prediction (EIP) via on-the-fly parametric transformation of ME candidates. The approach is validated in the particular case of ST. We show an efficient technique to compute the ST parameter and distortion in a single-step. The framework is fully integrated in H.264/AVC encoder in the context of RD optimization. Results show significant improvements over the baseline encoder, with BD-rate gains in average of 6% and up to 19%.

The approach can be used in any video codec that uses MC prediction, which includes most conventional state-of-the-art encoders. Currently the EIP approach is being investigated in the context of the HEVC draft; several considerations need to be addressed in this case mainly concerning the larger size and different sub-partition of the MB used for inter-prediction, and the inclusion of different coding tools such as the merge mode. Also, the approach is being improved by investigating specific entropy coders designed to encode the parameters, and by exploring other parametric transformations.

REFERENCES

[1] T.Wiegand, G. J. Sullivan, G. Bjontegaard and A. Luthra, “Overview of the H.264/AVC video coding standard”, *IEEE Transactions on Circuits and Systems for Video Technology*, v. 13, n.7, pp. 560-576, 2003.

[2] B. Girod, “Motion-compensating prediction with fractional-pel accuracy”, *IEEE Transactions on Communications*, v. 41, n. 4, pp. 604-612, 1993.

[3] ITU-T. Itu-t recommendation H.264, Advanced Video Coding for Generic Audiovisual Services, ITU-T, 2005.

[4] M. Haller, A. Krutz and T. Sikora, “Robust global motion estimation using motion vectors of variable size blocks and automatic motion model selection”, *IEEE International Conference on Image Processing*, pp.737-740, 2010.

[5] ISO. ISO/IEC 14496-2:2004 - Information technology - Coding of audio-visual objects - Part 2: Visual, 2004.

[6] M. Kemal Gullu, “Block motion estimation using luminance transform based matching” *IEEE 16th Signal Processing, Communication and Applications Conference*, pp. 1-4, 2008.

[7] You Zhou, Xiaoyan Sun, Hong Bao and Shipeng Li, “Weighted motion estimation for efficiently coding scene transition video”, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, v. 3, pp. iii-361-4, 2004.

[8] J. M. Boyce, “Weighted prediction in the H.264/MPEG AVC video coding standard,” *International Symposium on Circuits and Systems*, v.3, pp. iii-789-92, 2004.

[9] Joint Model (JM), “H.264/AVC Reference Software”, *Artech House Inc.*, http://iphome.hhi.de/suehring/tml/download

[10] “Joint Call for Proposals on Video Compression”, ISO/ IEC JTC1/SC29 /WG11/ N11113, ITU-T Q6/16 document, VCEG-AM91, 2010.

[11] C.-M. Fu, C.-Y. Chen, Y.-W. Huang and S. Lei, “Sample adaptive offset for HEVC”, *IEEE 13th International Workshop on Multimedia Signal Processing*, pp.1-5, 2011.

[12] Gisle Bjontegaard, “Improvements of the BD-PSNR model”, ITU-T SG16/Q6, 35th VCEG Meeting, Doc.VCEG-AI11, 2008.