Recursive Prediction for Joint Spatial and Temporal Prediction in Video Coding

Fatih Kamisli, *Member, IEEE*

***Abstract*—Video compression systems use prediction to reduce redundancies present in video sequences along the temporal and spatial dimensions. Standard video coding systems use either temporal or spatial prediction on a per block basis. If temporal prediction is used, spatial information is ignored. If spatial prediction is used, temporal information is ignored. This may be a computationally efficient approach, but it does not effectively combine temporal and spatial information. In this letter, we provide a framework where available temporal and spatial information can be combined effectively to perform joint spatial and temporal prediction in video coding. Experimental results obtained from one sample realization of this framework show its potential.**

***Index Terms*—Markov processes, motion compensation, video coding.**

I. Introduction

TYPICAL video sequences contain a significant amount of redundant information, along the temporal and spatial dimensions. Video compression is accomplished by exploiting these redundancies. Many video compression systems exploit these redundancies with a block-based approach consisting of two steps. In the first step, a block of pixels is predicted from previously coded pixels. In the second step, the block of prediction error pixels is transform-coded.

In the prediction step, previously coded pixels from either the temporal or spatial dimension are used. Standard video coding systems use either temporal or spatial prediction on a per block basis [1], [2]. In many instances, temporal correlation is much higher than spatial correlation and prediction is generated using only temporal information and the spatial information is ignored. In some instances, temporal information may not be available or it may be poor, and prediction is generated using only spatial information, ignoring any temporal information. In summary, if temporal prediction is used, available spatial information is ignored. If spatial prediction is used, temporal information is ignored. This may be a computationally efficient approach, but it is clearly suboptimal as it does not effectively combine temporal and spatial information.

Manuscript received February 16, 2014; revised March 16, 2014; accepted March 24, 2014. Date of publication March 28, 2014; date of current version April 07, 2014. This work was supported in part by Grant 113E516 of TUBITAK, the Scientific and Technological Research Council of Turkey. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yao Zhao.

The author is with the Department of Electrical and Electronics Engineering, Middle East Technical University, Ankara, Turkey.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/LSP.2014.2314215

In this letter, we provide a framework in which available temporal and spatial information can be combined effectively to perform joint spatial and temporal prediction in video coding. We use a spatio-temporal Markov process to model the spatial and temporal information in video sequences. This approach leads to a recursive spatio-temporal prediction algorithm, which generalizes the conventional temporal and spatial prediction methods.

The remainder of the letter is organized as follows. Section II reviews related previous research. Section III presents the proposed spatio-temporal prediction approach. Section IV discusses a sample realization of the proposed approach and presents experimental results obtained with this realization. Finally, Section V summarizes the letter and discusses future research directions.

II. Previous Research

For prediction in video coding, previously coded pixels from either the temporal or spatial dimension are used. For prediction in the temporal dimension (inter prediction), the motion-compensated prediction method is used. In this method, it is assumed that adjacent frames differ due to translational motion of objects or camera, and prediction is performed by copying a block (determined by accounting for the motion) from a previously coded frame. For prediction in the spatial dimension (intra prediction), previously coded neighboring pixels of the block are used. The intra prediction methods in recent video compression standards such as H.264 [1] or HEVC [2] perform spatial prediction by copying these neighbor pixels along one of many predefined directions inside the block.

Conventional video coding systems decide between inter and intra prediction on a per block basis. They use either temporal or spatial prediction but do not combine the two. The literature contains a number of proposals for combining temporal and spatial prediction, which we review below.

Seiler *et al.* perform spatio-temporal prediction with a two-step algorithm [3], [4], [5]. The first step is conventional motion-compensated prediction. The second step is a computationally intensive refinement step, which modifies the prediction from the first step. Their initial refinement algorithm [3] requires many iterations to converge. Their subsequent refinement algorithms [4], [5] reduce the number of iterations but still require excessive amount of computations rendering these algorithms prohibitive for practical video coding systems.

More feasible spatio-temporal prediction algorithms are proposed in [6], [7], [8]. In these proposals, conventional temporal

and spatial prediction methods are used to obtain two predictions separately, which are then combined with weighted averaging to obtain the final prediction. In [6], the proposal is implemented within H.264, and any inter prediction mode available in H.264 can be combined with any intra prediction mode available in H.264. The chosen inter and intra prediction modes are explicitly coded by the encoder together with the weights used for averaging. Reported coding gains are typically below $\%1$. This system is simplified in [7] by allowing the combination of only one particular inter and intra prediction mode with a pre-determined weight. Reported coding gains are similar to those of [6].

While the approaches in [6], [7] average the spatial and temporal predictions using the same weights for all block pixels, the weights are changed for each pixel in [8]. For temporal prediction, the prediction error is uniformly distributed in the block. For spatial prediction, however, the prediction error is smaller for pixels near the prediction boundary than for pixels away from the boundary. Based on this observation, different weights for each pixel are used when averaging temporal and spatial predictions in [8].

III. Joint Spatio-temporal Prediction Based on a Spatio-Temporal 3-D Markov Process Model

In [6], [7], [8], conventional temporal and spatial prediction methods are used to obtain two *separate* predictions, which are then combined by weighted averaging to obtain the final spatio-temporal prediction. This approach does *not jointly* utilize the available temporal and spatial information. Consider Fig. 1, where pixels $u(1,1)$, $u(1,2)$ etc. are predicted by averaging their separately obtained temporal and spatial predictions. When predicting pixel $u(1,1)$, the temporal prediction $u_t(1,1)$ is combined with the spatial prediction $u(1,0)$. When predicting pixel $u(1,2)$, the temporal prediction $u_t(1,2)$ is combined with the spatial prediction $u(1,0)$, however, the previously estimated pixel $\hat{u}(1,1)$ is closer to $u(1,2)$ and has statistically more reliable spatial information than $u(1,0)$. The approach we present here accounts for this sub-optimality and performs optimal *joint* spatio-temporal prediction based on the random process model we assume for the video signal.

The proposed joint spatial and temporal prediction framework in this letter is based on modeling video pixels in a local temporal and spatial neighborhood with a stationary spatio-temporal 3-D Markov process, which can be represented with the following recursive relationship

$$u(i,j) = \rho_1 u(i-1,j) + \rho_2 u(i-1,j-1) + \rho_3 u(i,j-1) + \rho_t u_t(i,j) + e(i,j). \quad (1)$$

Here, $u(i,j)$ represent image pixels in the current frame and $u_t(i,j)$ represent their motion-compensated reference pixels in the previously coded frame. It is assumed that $u(i,j)$ and $u_t(i,j)$ are zero-mean and unit variance, and $e(i,j)$ form a zero-mean white-noise process independent of pixels. This 3-D Markov process model assumes that each pixel is conditionally uncorrelated with all pixels, given its left, upper-left, upper spatial neighbor pixels and its motion-compensated temporal reference pixel (See Fig. 2). Note that similar 1-D or 2-D Markov processes have been used in image/video compression such as in the derivation of Discrete Cosine Transform (DCT) [9], in the recent development of Odd Type-3 Discrete Sine Transform (ODST-3) [10], or our recursive intra prediction approach [11], which indeed inspired this work.

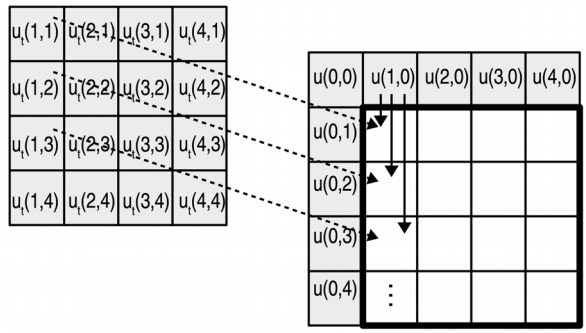

Fig. 1. Averaging of separately obtained temporal and spatial predictions according to [6], [7], [8]. Gray pixels on left indicate motion-compensated prediction block used for temporal prediction and gray pixels on right indicate spatial neighbors of block used for spatial prediction. When predicting block pixel $u(1,1)$, the temporal prediction $u_t(1,1)$ is combined with the spatial prediction $u(1,0)$. When predicting block pixel $u(1,2)$, the temporal prediction $u_t(1,2)$ is combined with the spatial prediction $u(1,0)$ etc.

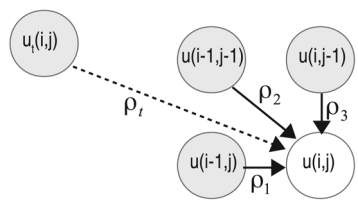

Fig. 2. Graphical representation of assumed spatio-temporal Markov process.

The Minimum-Mean-Square-Error (MMSE) estimate of a random variable is its conditional expectation given available observations. In our joint temporal and spatial prediction problem, the observations are the previously encoded temporal and spatial neighbor pixels of the block, and the MMSE estimate $\hat{u}(i,j)$ of any zero-mean block pixel can be obtained by computing its conditional expectation $E[u(i,j)|\mathbf{n}]$ (where $\mathbf{n}$ represents all available previously encoded temporal and spatial neighbor pixels of the block, i.e. gray pixel in Fig. 1.)

From Equation (1), the MMSE estimate $\hat{u}(1,1)$ can be easily determined using its previously encoded temporal and spatial neighbor pixels as

$$\hat{u}(1,1) = E[u(1,1)|\mathbf{n}] = \rho_1 u(0,1) + \rho_2 u(0,0) + \rho_3 u(1,0) + \rho_t u_t(1,1). \quad (2)$$

Similarly, from Equation (1), the estimate $\hat{u}(1,2)$ can be determined as

$$\hat{u}(1,2) = E[u(1,2)|\mathbf{n}] = \rho_1 u(0,2) + \rho_2 u(0,1) + \rho_3 \hat{u}(1,1) + \rho_t u_t(1,2) \quad (3)$$

where its previously encoded temporal and spatial neighbor pixels $u(0,2)$, $u(0,1)$, $u_t(1,2)$ and the previously computed estimate $\hat{u}(1,1)$ are used.

Continuing the computation of the estimates in a causal order (i.e. from top to bottom starting with the left row, or from left to

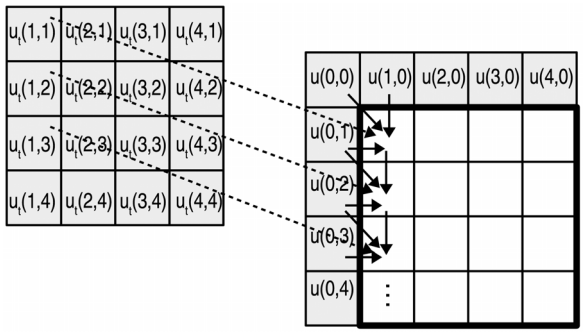

Fig. 3. Graphical representation of the obtained spatio-temporal prediction algorithm. Each pixel is predicted using its temporal reference pixel and its three spatial neighbors, which can be previously encoded neighbor pixels of the block or previously computed predictions.

right starting with the top row), the estimate $\hat{u}(i,j)$ of any block pixel can be determined from Equation (1) as

$$\hat{u}(i,j) = \rho_1\hat{u}(i-1,j) + \rho_2\hat{u}(i-1,j-1) + \rho_3\hat{u}(i,j-1) + \rho_t\hat{u}_t(i,j) \tag{4}$$

where $\hat{u}(i-1,j)$, $\hat{u}(i-1,j-1)$, $\hat{u}(i,j-1)$ and $\hat{u}_t(i,j)$ are either previously encoded known temporal or spatial neighbors of the block or previously computed estimates.

In summary, the MMSE estimate of the block pixels based on the assumed stationary spatial-temporal 3-D Markov process becomes a recursive spatio-temporal prediction algorithm, which is summarized in Fig. 3.

In the development of the recursive prediction algorithm, we assumed that video pixels $u(i,j)$ and $u_t(i,j)$ are zero-mean. To comply with that assumption, the mean is subtracted out from the previously encoded temporal and spatial neighbor pixels of the block prior to the application of the recursive spatio-temporal prediction algorithm and then the mean is added to the obtained zero-mean estimates of the block pixels. In our experiments, we assume that all previously encoded temporal and spatial neighbor pixels and the estimated block pixels have the same mean, which we compute by averaging all pixels of the temporal reference block.

It is worth noting here that the conventional temporal and spatial prediction methods are special cases of the proposed spatio-temporal recursive prediction algorithm. In particular, using $\rho_1 = \rho_2 = \rho_3 = 0$ and $\rho_t = 1$ produces the conventional temporal prediction method. Using $\rho_1 = 1$ and $\rho_2 = \rho_3 = \rho_t = 0$ produces the conventional spatial prediction method with horizontal copying of block neighbor pixels. Using nonzero parameters produces joint spatial and temporal prediction.

The parameters for MMSE estimation with the recursive prediction algorithm are determined by the cross-correlation of the pixels and are given by

$$\begin{bmatrix}\rho_1\\ \rho_2\\ \rho_3\\ \rho_t\end{bmatrix} = \begin{bmatrix}1 & R_v & R_x & R_{h,t}\\ R_v & 1 & R_h & R_{d,t}\\ R_x & R_h & 1 & R_{v,t}\\ R_{h,t} & R_{d,t} & R_{v,t} & 1\end{bmatrix}^{-1} \cdot \begin{bmatrix}R_h\\ R_d\\ R_v\\ R_t\end{bmatrix} \tag{5}$$

where R_h represents horizontal, R_v represents vertical, R_d and R_x diagonal, R_t temporal, and $R_{h,t}$, $R_{v,t}$ and $R_{d,t}$ represent horizontal-temporal, vertical-temporal and diagonal-temporal correlations. To be more precise (see Fig. 2),

$$\begin{aligned}R_h &= E[u(i,j)u(i-1,j)] = E[u(i,j-1)u(i-1,j-1)]\\ R_v &= E[u(i,j)u(i,j-1)] = E[u(i-1,j)u(i-1,j-1)]\\ R_d &= E[u(i,j)u(i-1,j-1)]\\ R_x &= E[u(i-1,j)u(i,j-1)]\\ R_t &= E[u(i,j)u_t(i,j)]\\ R_{h,t} &= E[u(i-1,j)u_t(i,j)]\\ R_{v,t} &= E[u(i,j-1)u_t(i,j)]\\ R_{d,t} &= E[u(i-1,j-1)u_t(i,j)].\end{aligned}$$

Notice that these correlations are likely to change significantly in different temporal and spatial regions of typical video sequences. Thus, for best video compression performance, the recursive spatio-temporal prediction framework proposed in this section needs to be realized with an adaptive and rate-distortion optimal design. Notice also that the obtained recursive spatio-temporal prediction algorithm is independent of prediction block-size and scales easily to any block-size.

IV. Sample System Realization and Experimental Results

We first present in Section IV-A a simple realization of the proposed recursive spatio-temporal prediction framework. Experimental results obtained with this system are presented in Section IV-B.

A. Sample System Realization

The proposed recursive spatio-temporal prediction framework can be realized in many different ways in practical video coding systems. One approach is to store predetermined groups of prediction parameters ρ_1, ρ_2, ρ_3 and ρ_t in a table at both encoder and decoder, and let the encoder transmit an index to indicate a group from the table for each block. For best video compression performance, a rate-distortion optimal realization is desirable. For the experimental results in this letter, the following realization is used. It is assumed that the block to be predicted and its temporal reference block (i.e. its motion-compensated reference block) have similar spatial characteristics and therefore the spatial correlations R_h, R_v, R_d and R_x are estimated by both the encoder and decoder from the temporal reference block. It is also assumed that the spatio-temporal correlations $R_{h,t}$, $R_{v,t}$ and $R_{d,t}$ are separable, i.e. $R_{h,t} = R_h \cdot R_t$, $R_{v,t} = R_v \cdot R_t$ and $R_{d,t} = R_d \cdot R_t$. The value of the temporal correlation R_t is determined by a table (stored at both encoder and decoder) depending on the block-size for motion estimation (i.e. inter prediction macroblock mode, See Table I). With these assumptions, all correlations required to compute the spatio-temporal prediction parameters from Equation 5 are determined, and the spatio-temporal prediction parameters are computed at both the encoder and decoder.

The used R_t parameters in Table I were determined offline from training sequences, which are not included in the experiments. Due to assumed separability of the correlation along the temporal dimension, if $R_t = 1$, the proposed spatio-temporal

TABLE I
USED TEMPORAL CORRELATION R_t FOR EACH INTER MACROBLOCK MODE

Mode :	P16x16	P16x8	P8x16	P8x8
R_t :	0.92	0.92	0.92	0.96

prediction reduces to conventional temporal prediction. If R_t is much smaller than 1, then the prediction weights of the temporal neighbors become too small. The online estimation of the spatial correlation parameters from the temporal reference block were performed using sample averages. For example, R_h was estimated as (where μ_t is the mean of the block)

$$\hat{R}_h = \frac{1}{N^2} \sum\nolimits_{i,j=1}^{N} (u_t(i,j) - \mu_t)(u_t(i-1,j) - \mu_t). \quad (6)$$

We have implemented the described realization of the recursive spatio-temporal prediction approach into the reference software of H.264 by modifying the inter macroblock coding modes of luma pictures. Coding of chroma pictures or intra macroblock modes of luma pictures were not modified. The modification of the inter macroblock coding modes *P16x16, P16x8, P8x16* and *P8x8* is as follows. For each partition of the macroblock that has a separate motion vector associated with it, a 1-bit flag is used to indicate whether this partition is predicted with conventional inter prediction or with the recursive spatio-temporal prediction realization discussed above. In *P8x8* mode, a single 1-bit flag is used although each 8×8 block can be subdivided further. The encoder decides on the value of the 1-bit flag in each partition with rate-distortion optimization [12]. (Note that the motion estimation algorithm is not modified, only the prediction process is modified.)

To summarize, the major steps for decoding a macroblock in the modified system are as follows. First, the macroblock type and the motion vector for each of its partitions are decoded and the motion-compensated temporal reference is obtained for the macroblock. Next, a 1-bit flag for each partition is decoded. Finally, each 4×4 block of the macroblock is reconstructed in two steps. In the first step, the 4×4 block is predicted, depending on the decoded flag of its partition, using either conventional temporal prediction, or the described realization of recursive spatio-temporal prediction. In the second step, residual data of the 4×4 block, obtained from decoded transform coefficients, is added to the prediction in the first step.

B. Experimental Results

We present experimental results to show achievable coding gains with the recursive spatio-temporal prediction realization discussed in Section IV-A, which we call RSTP hereafer. Compression results of RSTP are compared with those of the default H.264 reference software using the Bjontegaard-Delta bitrate (BD-BR) [13] metric, which uses the PSNR of the luma pictures and the total bitrate (including bitrate of 1-bit flag) of luma and chroma pictures. The BD-BR metric roughly gives the average percentage bitrate saving of one system with respect to another system averaged over a range of picture qualities. The range of picture qualities are determined by encoding a sequence with four different Quantization Parameters (QP) of the H.264 standard, which were 30, 25, 20, 15 in our experiments and typically correspond to a range of 32 dB to 45 dB.

TABLE II
: BD-BR BITRATE SAVINGS (%) OF RSTP WITH RESPECT TO H.264

Sequence Name	Bitrate savings(%)
bridge-close-qcif-30	1.51
basket-cif-30	1.71
bridge-close-cif-30	1.80
coastguard-cif-30	3.45
football-cif-30	1.90
BQMall-832x480-60	2.14
RaceHorses-832x480-30	1.34
vidyo1-720p-60	2.13
vidyo3-720p-60	2.19
vidyo4-720p-60	2.63
Kimono1-1920x1080-24	2.34

Some important encoder configuration parameters common to both systems are as follows. Baseline profile is used and the first frame is coded as an I-frame and the remaining as P-frames. All macro-block modes are enabled, and the best mode is determined using rate-distortion optimized mode selection. Motion vectors are of quarter-pel resolution and are searched with the fast EPZS algorithm.

The achieved coding gains of RSTP with respect to the default H.264 reference software are shown in Table II. It can be seen that consistent coding gains are achieved with upto $\%3.45$ average bitrate savings.

To compare the complexities of the systems, their average encoding and decoding times are provided. The average encoding and decoding times of RSTP are 199% and 190% of those of H.264. Note that the modifications in RSTP have not been programmed considering the running times by any means, and with proper consideration, the increase in encoding and decoding times are expected to be significantly less.

V. CONCLUSION

Standard video coding systems switch between temporal and spatial prediction on a per block basis. If temporal prediction is used, spatial information is ignored. If spatial prediction is used, temporal information is ignored. This may be a computationally efficient approach, but it does not effectively combine temporal and spatial information. In this letter, we presented a framework in which available temporal and spatial information are combined with a recursive spatio-temporal prediction framework.

Experimental results obtained with one sample realization of this framework showed promising coding gains and there are many aspects that can be improved. One aspect is the assumed separability of the spatio-temporal correlations. We have not checked the accuracy of this assumption and more accurate models of spatio-temporal correlation are likely to improve coding gains. Another aspect is to use adaptive temporal correlations, R_t. Using the same R_t values for all sequences, frames, and coding bitrates is not efficient, and adaptive R_t values are likely to improve results. Investigation of the proposed approach in bi-predictively coded macroblocks is also of interest. In summary, while this letter introduced the recursive spatio-temporal prediction framework, future research is needed for more efficient and successful application of this framework in video coding systems.

REFERENCES

[1] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, pp. 560–576, Jul. 2003.

[2] G. Sullivan, J. Ohm, W.-J. Han, T. Wiegand, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[3] J. Seiler and A. Kaup, "Spatio-temporal prediction in video coding by spatially refined motion compensation," in *15th IEEE Int. Conf. Image Processing, ICIP 2008*, Oct 2008, pp. 2788–2791.

[4] J. Seiler, H. Lakshman, and A. Kaup, "Spatio-temporal prediction in video coding by best approximation," in *Picture Coding Symp., 2009 PCS*, May 2009, pp. 1–4.

[5] J. Seiler and A. Kaup, "Multiple selection approximation for improved spatio-temporal prediction in video coding," in *2010 IEEE Int. Conf. Acoustics Speech and Signal Processing (ICASSP)*, Mar. 2010, pp. 886–889.

[6] K. Andersson, "Combined intra inter prediction coding mode," ITU-T SG16/Q6 (VCEG), Doc. VCEGAD11 Oct. 2006.

[7] J. Xin, K. N. Ngan, and G. Zhu, "Combined inter-intra prediction for high definition video coding," in *Picture Coding Symp. (PCS)*, Nov. 2007.

[8] R. Cha, O. C. Au, X. Fan, and F. Zou, "An efficient combined inter and intra prediction scheme for video coding," in *APSIPA Annu. Summit and Conf. (ASC)*, Oct. 2011.

[9] M. Flickner and N. Ahmed, "A derivation for the discrete cosine transform," *Proc. IEEE*, vol. 70, no. 9, pp. 1132–1134, Sep. 1982.

[10] C. Yeo, Y. H. Tan, Z. Li, and S. Rahardja, "Mode-dependent transforms for coding directional intra prediction residuals," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 4, pp. 545–554, April 2012.

[11] F. Kamisli, "Intra prediction based on markov process modeling of images," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3916–3925, 2013.

[12] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE, Signal Process. Mag.*, vol. 15, no. 6, pp. 74–90, Nov 1998.

[13] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," ITU-T Q.6/SG16, VCEG-M33 2001.