LOSSLESS VIDEO COMPRESSION USING A SPATIO-TEMPORAL OPTIMAL PREDICTOR

Stefano Andriani, Giancarlo Calvagno, Gian Antonio Mian

Department of Information Engineering, University of Padova
Via Gradenigo, 6/B - 35131 Padova - Italy
e-mail: {mutley,calvagno,mian}@dei.unipd.it

ABSTRACT

Lossless video compression is a novel research area, but it is gaining widespread importance. As an example, in digital cinema the post-production chain requires all the information captured by digital cameras, and any data loss is not tolerated. On the other hand, the camera sensors size is expected to grow up to 4k × 2k pixels at 10 bit per pixel per component, and cameras output frame-rate up to 150 fps. This enourmous amount of data asks for efficient lossless compression techniques.

In this paper we propose a novel compression algorithm based on an optimal predictor which exploits the temporal correlation. This solution provides an improvement of the compression ratio, but the resulting algorithm is computationally demanding. An alternative method reducing the overall complexity is presented.

1. INTRODUCTION

Lossless compression algorithms can be divided in two major categories: prediction based coding and transform based coding.

In prediction coding, the pixels of the image are predicted using the spatial correlation, and the residual image (or prediction error image) is entropy coded. If the predictor is accurately designed, the residual image entropy is lower than the original image entropy, and consequently, the compression ratio improves.

Transform coding applies a reversible transformation to the image. The resulting image is divided into four or more sub-bands. The sub-band located at the lower frequencies contains most of the signal energy, while the others, at higher frequencies, include a small part of image energy. This kind of coding is tipically used when working in lossy-to-lossless mode, as in JPEG2000.

The proposed scheme includes the temporal information in the predictor equations. As a result, the predictor can estimate the existing correlation between spatial and temporal adjacent pixels, and if a scene change occours it does not keep into account the pixels belonging to the previous frame, because the two consecutive frame are loosely correlated.

Moreover, the spectral prediction correction introduced in [1] was successfully used in order to improve the performance.

The paper is structured as follows: Section 2 presents the GLICBAWLS algorithm which is our starting coding scheme. In Section 3 we present the proposed algorithm and its simplified version. In Section 4 we compare the proposed algorithms with respect to WLOPT-3D algorithm [2, 3] and the lossless standard coder JPEG-LS [4]. Finally, in Section 5 we report the conclusions.

Work carried out within the 6-th Framework Programme IST-MetaCamera Project: 2003-506969

2. THE GLICBAWLS ALGORITHM

GLICBAWLS is an acronym for Grey Level Image Compression By Adaptive Weighted Least Squares. It is a prediction based coder introduced by Meyer and Tischer in [5].

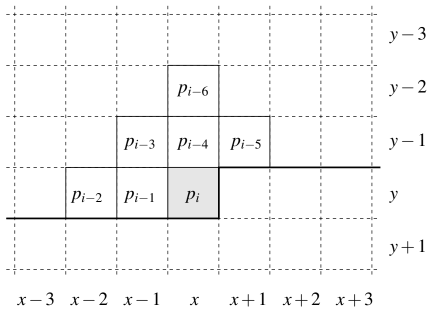

Figure 1: Pixel notation used by the proposed algorithms in the current frame.

Glicbawls constructs the adaptive optimal linear predictor of the current pixel at position (x_C, y_C) using the six neighbor causal pixels with Manhattan distance lower than or equal to two (in the following referred to as P6), see Fig. 1. New weights are calculated for each pixel of the image, taking into account pixels already coded.

Given the matrix $\mathbf{A}_i = \mathbf{p}_i\,\mathbf{p}_i^T$ of pixel i, where $\mathbf{p}_i = [p_{i-1}, \ldots, p_{i-6}]^T$ according to Fig. 1 notation, and vector $\mathbf{b}_i = p_i \mathbf{p}_i$, for each component R, G or B we update the autocorrelation matrix as

$$\mathbf{A_C} = \sum_{i=1}^{N} 0.8^{|x_C - x_i| + |y_C - y_i|} \mathbf{A_i} \tag{1}$$

and the vector

$$\mathbf{b_C} = \sum_{i=1}^{N} 0.8^{|x_C - x_i| + |y_C - y_i|} \mathbf{b_i} \tag{2}$$

where the weight $0.8^{|x_C - x_i| + |y_C - y_i|}$ is used to decrease the influence of the farthest pixels from the current position (x_C, y_C). In equations (1) and (2), N is the number of previously coded pixels. The prediction coefficients w_i are adaptively computed solving the linear system

$$\mathbf{A_C}\mathbf{w} = \mathbf{b_C}. \tag{3}$$

Therefore, the predicted pixel $\hat{p}_i$ is computed according to

$$\hat{p}_i = \sum_{k=1}^{6} w_k \, p_{i-k} \tag{4}$$

where p_{i-k} is the R,G or B value of the $(i-k)$-th pixel in the P6 set.

Glicbawls scheme is quite different with respect to the others prediction based algorithm. In fact, it does not code the residual image, but it uses the prediction errors to estimate the variance of a modified Student distribution. The actual value is then coded in a bit-plane mode using a binary arithmetic coder starting from the MSB (More Significant Bit) down to the LSB (Least Significant Bit). The "zero" probability is calculated by integrating the modified Student distribution centered on the predicted pixel value [5].

Since each frame is extended to a constant value outside the image boundaries, the decoder can replicate the coder operations without requiring that the predictor coefficients are sent as side-information.

This algorithm is computational demanding for two reasons: firstly, the optimal predictor is obtained inverting a 6×6 matrix for each pixel and for each colour band; secondly, the "zero" probability computation, using the estimated modified Student distribution, is not a trivial operation because it requires the numerical computation of an integral.

3. SCALAR PREDICTION USING TEMPORAL CORRELATION

The algorithm proposed in Section 2 uses only the spatial correlation and it is effective in still image coding. However, for video sequences the temporal correlation is very useful to increase the compression ratio, because consecutive frames are usually highly correlated. However, temporal correlation is useless when scene changes accour. In this case, all the algorithms based upon motion estimation are not efficient, because the estimated motion vectors estabilish an erroneous relationship between two indipendent frames. In literature many researchers proposed several algorithms to prevent the compression ratio reduction due the scene changes, but they usually have high computational cost and are used to switch from inter-frame coding to intra-frame coding.

In our work we aim at developing a new compression technique which is not only able to exploit the temporal correlation between adjacent frames, but also it is roboust against possible scene changes.

To this purpose, we introduce the temporal information inside the spatial optimal predictor presented in Section 2 by including in the predictor the pixels of the previous frame inside a 3×3 matrix centered at the same position of the predicted pixel.

3.1 Original version

Consider the vector $\mathbf{p}_i$ defined in Section 2. It contains the spatial causal pixels near to the current one. The temporal correlation is introduced by adding to $\mathbf{p}_i$ the nine pixels belonging to the 3×3 block (according to the notation of Fig. 2) of the previous frame centered at the same position of the predicted pixel

$$\mathbf{p}_i = [p_{i-1} \ \cdots \ p_{i-6} \mid p_{i-7} \ \cdots \ p_{i-15}]^T \tag{5}$$

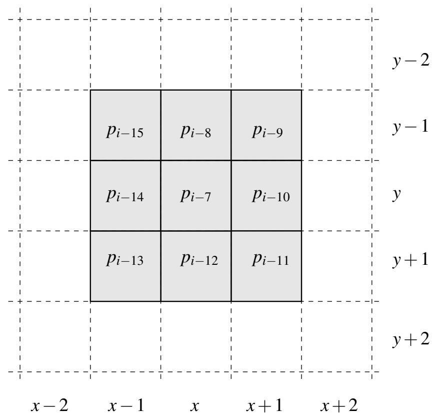

Figure 2: Pixel notation using by the LPOSTC and the LPOSTC-RGPC algorithms in the previous frame.

where the first six components contain the current frame pixel values, and the last nine components contain the previous frame pixel values. The resulting autocorrelation matrix update, $\mathbf{A}_i$, for a given colour band

$$\mathbf{A}_i = \mathbf{p}_i \mathbf{p}_i^T = \left[\begin{array}{c|c} \mathbf{A}_{i,11} & \mathbf{A}_{i,12} \\ \hline \mathbf{A}_{i,21} & \mathbf{A}_{i,22} \end{array} \right] \tag{6}$$

is a block diagonal symmetric matrix, where the four blocks $\mathbf{A}_{i,\cdot\cdot}$ have the following meanings:

$\mathbf{A}_{i,11}$ reflects the 6×6 spatial correlation for the current frame

$$\mathbf{A}_{i,11} = \begin{bmatrix} p_{i-1}^2 & p_{i-1}p_{i-2} & \cdots & p_{i-1}p_{i-6} \\ p_{i-1}p_{i-2} & p_{i-2}^2 & \cdots & p_{i-2}p_{i-6} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i-1}p_{i-6} & p_{i-2}p_{i-6} & \cdots & p_{i-6}^2 \end{bmatrix}; \tag{7}$$

$\mathbf{A}_{i,22}$ reflects the 9×9 spatial correlation for the previous frame

$$\mathbf{A}_{i,22} = \begin{bmatrix} p_{i-7}^2 & p_{i-7}p_{i-8} & \cdots & p_{i-7}p_{i-15} \\ p_{i-7}p_{i-8} & p_{i-8}^2 & \cdots & p_{i-8}p_{i-15} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i-7}p_{i-15} & p_{i-8}p_{i-15} & \cdots & p_{i-15}^2 \end{bmatrix}; \tag{8}$$

$\mathbf{A}_{i,12}$ and $\mathbf{A}_{i,21}$ reflect both spatial and temporal correlation between adjacent frame pixels

$$\mathbf{A}_{i,12} = \mathbf{A}_{i,21}^T = \begin{bmatrix} p_{i-1}p_{i-7} & p_{i-1}p_{i-8} & \cdots & p_{i-1}p_{i-15} \\ p_{i-2}p_{i-7} & p_{i-2}p_{i-8} & \cdots & p_{i-2}p_{i-15} \\ \vdots & \vdots & \ddots & \vdots \\ p_{i-6}p_{i-7} & p_{i-6}p_{i-8} & \cdots & p_{i-6}p_{i-15} \end{bmatrix}. \tag{9}$$

We solve three linear systems, one for each colour band,

$$\mathbf{A}_{C,b} \mathbf{w}_b = \mathbf{b}_{C,b} \tag{10}$$

and the resulting prediction coefficients are used to compute the predictions

$$\hat{p}_{i,b} = \sum_{k=1}^{15} w_{k,b} \, p_{i-k,b} \tag{11}$$

where $b = \{R, G, B\}$. The residual images are bit-plane coded using a binary arithmetic coder.

We call this algorithm Linear Prediction Over Spatial Temporal Correlation (LPOSTC).

3.2 Low computational cost version

In order to simplify the proposed algorithm, we decided to reuse the prediction coefficients calculated for the green band in the prediction used for the red and for the blue bands. Therefore, the solution of the system for the green band

$$\mathbf{A}_{C,G}\mathbf{w}_G = \mathbf{b}_{C,G}$$

provides the prediction coefficients for the three equations

$$\begin{aligned} \hat{p}_{i,R} &= \sum_{k=1}^{15} w_{k,G}\, p_{i-k,R} \\ \hat{p}_{i,G} &= \sum_{k=1}^{15} w_{k,G}\, p_{i-k,G} \\ \hat{p}_{i,B} &= \sum_{k=1}^{15} w_{k,G}\, p_{i-k,B}\,. \end{aligned} \quad (12)$$

In this way, we reduce three times the prediction computational cost of the original algorithm, but we have an optimal prediction only for the green band. For the red and the blue bands we only obtain a sub-optimal prediction. This assumption works considering that the three bands have the same behaviour and the same correlation between spatial and temporal neighbour pixels. We call this algorithm Linear Prediction Over Spatial Temporal Correlation Reusing Green Prediction Coefficients (LPOSTC-RGPC).

3.3 On the use of motion estimation

Motion estimation is based on the idea that a block in the current frame could be matched with a neighbour block in the previous frame. The motion compensation algorithm selects the block which minimizes the MSE (Mean Square Error) between the current frame block and all possible blocks in a given searching area of the previous frame. The motion vector is defined as the position offset between the position of the block in the current frame and the position of its best match in the previous frame. Motion compensation is used in lossy coding of video sequences (i.e. MPEG-2 or H.264), and many researchers have suggested new low complexity methods to calculate the motion vectors. However, motion estimation have some intrinsic problems:

- when a scene change occours the founded matches give erroneous information;
- it is not able to exploit complex movements (i.e. zoom, rotations, or fast motion);
- the coder must save the motion vectors as side information. Consequently, the block size could not be too small, because a lot of motion vectors have to be estimated and saved. On the other hand, the block size could not be too large otherwise the motion vector is the average of all the movements belonging to a block.

The presented algorithm addresses these problems, but it does not fully exploit temporal correlation in fast motion sequences. In fact, the 3×3 block in the previous frame is only able to exploit small movement (less than or equal to 1 pixel), but it is a good compromise between efficiency and computational cost. If we use a 5×5 block we could exploit larger movement (less than or equal to 2 pixels), but we have to manage a 31^{st}-order predictor and, consequently, a 31×31 matrix has to be inverted for each prediction. Using the Cholesky factorization the computational cost of this operation is $\mathcal{O}(31^3/6)$, and it is 9 times greater with respect to the proposed solution which uses a 3×3 block.

4. RESULTS

The proposed algorithms are tested using several 8 bpp R, G and B colour sequences. The results are calculated as the average of the first ten frames (from frame 2 to frame 11) and reported in bit per pixel per component [bpppc]. The first frame is not considered because it is coded in intra-mode. In literature there are only a few papers on lossless video compression, so our comparisons are restricted to the lossless compression standard for still images (JPEG-LS [4]), and the WLOPT-3D algorithm which is based upon an optimal prediction which uses motion compensation [2].

Sequence	LPOSTC		P6	
		Corrected		Corrected
Susie	3.50	3.25	4.02	3.60
Football	4.65	4.29	5.24	5.03
Claire	2.15	2.23	2.46	2.39
Missa	3.46	3.48	3.81	3.75
Mobile	4.68	4.21	5.33	4.81
Tennis	4.67	3.52	5.43	4.08
Renata	4.26	3.22	5.17	3.65
Calendar	4.49	3.92	5.35	4.26
Flowers	4.16	3.63	4.42	3.73
Average	4.00	3.52	4.57	3.92

Table 1: Comparison of the proposed LPOSTC and P6 algorithms [bpppc].

Table 1 reports the first comparison between the pure spatial predictor (P6) and the spatio-temporal predictor (LPOSTC) without or with (column "Corrected") the spectral correction presented in [1]. The difference between these two algorithms is given by the temporal information inside the predictor. The compression of the LPOSTC algorithm is greater (on average) of 0.57 bpppc than the compression of the P6 algorithm for the version that does not use spectral correction, and of 0.40 bpppc for the spectral corrected version. For the last four sequences the P6 spectral corrected algorithm works better than the LPOSTC without any spectral correction. In these sequences there is a lot of movement and high frequency so the 3×3 block in the previous frame is too small to fully exploit the temporal correlation.

Sequence	LPOSTC		LPOSTC-RGPC	
		Corrected		Corrected
Susie	3.50	3.25	3.50	3.25
Football	4.65	4.29	4.67	4.30
Claire	2.15	2.23	2.30	2.32
Missa	3.46	3.48	3.71	3.73
Mobile	4.68	4.21	4.75	4.16
Tennis	4.67	3.52	4.76	3.74
Renata	4.26	3.22	4.35	3.48
Calendar	4.49	3.92	4.67	4.22
Flowers	4.16	3.63	4.48	3.97
Average	4.00	3.52	4.13	3.68

Table 2: Comparison of the LPOSTC and the LPOSTC-RGPC algorithms [bpppc].

In Table 2 we compare the original version LPOSTC and its simplified version LPOSTC-RGPC. For some sequences the simplified version achieved the same compression gain of the LPOSTC algorithms. However, on the average, we lose about 0.14 bpppc, but the predictor computation of the reduced complexity algorithm is three time faster than the original algorithms. Using the Cholesky factorization the complexity for the former algorithm is $\mathscr{O}(15^3/6)$ and for the latter algorithm is $\mathscr{O}(3 \times 15^3/6)$.

Sequence	LPOSTC		LPOSTC-RGPC		WLOPT-3D	
		Corr.		Corr.		Corr.
Tennis	4.67	3.52	4.76	3.74	4.95	4.38
Renata	4.26	3.22	4.35	3.48	4.61	4.12
Calendar	4.49	3.92	4.67	4.22	4.75	4.52
Flowers	4.16	3.63	4.48	3.97	4.60	4.32
Average	4.00	3.52	4.57	3.85	4.73	4.34

Table 3: Comparison of the proposed LPOSTC, LPOSTC-RGPC and the WLOPT-3D algorithms [bpppc].

Table 3 reports the comparison between the proposed algorithms with respect to WLOPT-3D [2, 3].

The first operation of the WLOPT-3D scheme is motion compensation, where each motion vector relative to each block is determined by an absolute difference minimization. Then, it predicts each pixel using a weighted linear combination of pixels in the current and previous frames, by taking into account motion information. Finally, a context-based Golomb-Rice coder is applied to the residual image. To reduce its complexity, the authors proposed to calculate the optimal prediction coefficients only if the prediction error is greater than a fixed threshold. However, this solution introduces a compression ratio loss, and the resulting algorithm has an image-dependent complexity.

The proposed algorithms perform better than WLOPT-3D up to 0.82 bpppc (comparing the spectral corrected results), and the gain achieved by the reduced-complexity algorithm (LPOSTC-RGPC) is about 0.5 bpppc. Moreover, the computational cost of the LPOSTC-RGPC algorithm is smaller with respect to the WLOPT-3D algorithm.

Sequence	LPOSTC		LPOSTC-RGPC		JPEG-LS
		Corr.		Corr.	
Tennis	4.67	3.52	4.76	3.74	5.78
Renata	4.26	3.22	4.35	3.48	5.41
Calendar	4.49	3.92	4.67	4.22	5.32
Flowers	4.16	3.63	4.48	3.97	4.95
Average	4.00	3.52	4.57	3.85	5.36

Table 4: Comparison of the proposed LPOSTC, LPOSTC-RGPC and the JPEG-LS algorithms [bpppc].

Algorithm \ Frame	53	54	55
P6	3.48	3.63	3.63
LPOSTC	3.04	3.67	3.09

Table 5: Behaviour of the LPOSTC algorithm when a scene changes occours [bpppc].

In Table 4 we compare the results obtained with respect to JPEG-LS. The proposed algorithms achieve better compression ratio, but JPEG-LS is a still image coder, so it is not able to exploit the temporal information. Moreover, we apply a spectral prediction error correction which introduces an improvement of about 0.5 bpppc, while JPEG-LS codes the three colour components, independently.

Finally, in Table 5 we report the behaviour of the LPOSTC algorithm when a scene change occours. In fact, the sequence Tennis has a scene change between the 53^{rd} and the 54^{th} frames. Coding the 53^{rd} frame the LPOSTC algorithm achieves a better compression ratio than the P6 algorithm, because it exploits the 52^{nd} temporal correlated frame. When the scene change occours, the LPOSTC algorithm uses, during the prediction, two loosely correlated frames and its performance is close to the P6 performance (only 0.04 bpppc difference). After that, the 54^{th} and the 55^{th} frames are temporally correlated and the LPOSTC algorithms well exploits this additional information. In this way, we proved the roboustness of the proposed algorithm against the actual problem of the scene changes.

5. CONCLUSIONS

In this paper we presented a new way to exploit the temporal correlation inside an adaptive optimal linear predictor. This novel idea tries to solve the problems introduced by motion estimation and compensation. Moreover, the proposed scheme does not need additional techniques to detect possible scene changes in order to switch from inter-frame to intra-frame coding.

The presented algorithms exploit both spatial and temporal correlation inside an optimal predictor. This solution gives a good compression ratio and it is roboust against the scene changes problem.

The reduced-complexity proposed algorithm is three times faster (if we only consider the predictor computation time) than the original one, and its results are only slightly worse than those of the original version. However, it gives a good trade-off between low computational load and achievable compression ratio.

REFERENCES

[1] S. Andriani, G. Calvagno, and G. Mian, "A Lossless Image Coding Technique Exploiting Spectral Correlation on the RGB space," in *Proc. EUSIPCO 2004*, Wien, Austria, Sept. 2004, pp. 1305–1308.

[2] D. Brunello, G. Calvagno, M. Durigon, and R. Rinaldo, "Optimal Weighted 3D Prediction for Lossless Video Coding," in *Proc. Non-linear Signal and Image Processing - NSIP 2003*, Grado, Italy, June 2003, pp. 89–93.

[3] D. Brunello, G. Calvagno, G. Mian, and R. Rinaldo, "Lossless compression of video using temporal information," *IEEE Trans. Image Processing*, vol. 12, no. 2, pp. 132–139, Feb. 2003.

[4] M. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Processing*, vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[5] B. Meyer and P. Tischer, "GLICBAWLS - Grey Level Image Compression By Adaptive Weighted Least Squares," in *Proc. Data Compression Conference (DCC 2001)*, Snowbird, Utah, USA, Mar. 2001, p. 503.