Temporal Stationarity Based Prediction Method For Lossless Video Coding

Mohit Vaishnav*
The LNM Institute of Information Technology
mohit.vaishnav@gmail.com

Dinesh Kumar Chobey
Th LNM Institute of Information Technology
dinesh.kumar@ieee.org

Anil Kumar Tiwari
Indian Institute of Technology, Jodhpur
akt@iitj.ac.in

ABSTRACT

This paper aims to propose a noble method to estimate the auto-regressive(AR) coefficients used by least-square(LS) based predictors. Estimation of this LS based predictors is computationally most complex process. This process requires a covariance matrix comprised of chosen causal pixels and also the inverse elements of the same matrix. Computational requirements of this process depends on the number of pixels for which the predictor is trained and also on the order of the predictor. Due to this high complexity, the predictor is not used practically although it provides a high compression ratio. Thus, an alternative algorithm, popularly known as LOPT-3D, was proposed in literature. However, the number of pixels required for the estimation of AR parameters are still large, and thereby, making it impracticable for real-time implementations. The proposed method overcomes this limitation by effectively making use of previously estimated AR parameters.

Keywords

Entropy, Least Square method, Motion compensation.

1. INTRODUCTION

Lossless compression offers the best of both the worlds, with mathematically identical content to an uncompressed file while simultaneously enjoying the benefits of a smaller file size in increased file storage versus uncompressed files, and the benefits of faster transportation of the compressed file. It has numerous applications such as tele-medicine, satellite imagery, medical imaging, 4G, remote sensing, radio nuclide imaging, mining satellite imaging, CT scanning and plenty more, although it offers a very low compression ratio.

A lot of work has been done in field of video compression which are as follows. In [7], the authors have suggested an adaptive scheme switching from temporal to spectral prediction and have proposed coding procedures of color video sequences considering both temporal and spectral redundancy. While in [8], a different predictor for inter band correlation exploitation was proposed. Another method, proposed in [3], works on block based feed forward type of prediction. This prediction method was employed in CALIC frame work to get compression. This method uses a set of pixels from the previous two frames while predicting the current pixel.

To estimate the level of activity in the prediction context of a pixel, a method was proposed in [9], which presents a context based predictive coding method for lossless compression of video. In this, the pixels are classified into bins and then Least Square (LS) based predictors are estimated for pixels belonging to each of the bins, based on the classified value of the level of activity. For minimizing the complexity of LS based predictor, a new method popularly known as LOPT-3D, was introduced in [2] which is a feedback type of LS based predictor and it estimates the AR parameters only for edged pixels. A pixel is declared to be edged if the amplitude of the prediction error at the previous location (in raster scan order) is larger than a preset threshold value.In [4], the authors had presented a scheme which estimates the AR parameters in advance using a set of training frames and later use them for prediction of next set of frames. In [11], the author have estimated the pixel based on its deviation from the causal pixels in the previous frame. These causal pixels are divided into bins, having fixed coefficients, on the basis of distance between current and causal pixel location.

Most of the algorithms mentioned above uses motion compensated frames. In literature, there has been an appreciable amount of work done to estimate the motion compensated frame. As, Hough transform was applied in [6], after first dividing each frame of the video sequence into blocks using the block matching algorithm. Blocks containing similar motion parameters, are estimated for each segment using several motion models and least-squares algorithms. In [5], control grid interpolation (CGI), which is a new class of motion compensation, has been used for video compression. In [1], an object based video compression system was developed which uses foreground motion compensation scheme and has application in the field of video archival and surveillance. In [10], a simple algorithm has been presented to find out the motion compensated frame, where they have used a pre-determined threshold value to estimate the motion compensated pixel in the previous frame.

This paper proposes a method which achieves nearly same compression as attained by LS based predictor with very low

*Corresponding author

http://dx.doi.org/10.1145/2683483.2683522

computational complexity. Unlike past methods, which evaluates AR parameters for every pixels of all the frames in the video sequence, this method efficiently uses the parameters estimated for a frame, in the prediction for a set of forthcoming frames without any computation. This reduces the cost and complexity associated with estimation of parameters to a great extent. Temporal redundancy has been exploited in deciding how the previously acquired parameters are to be used for the forthcoming frames.

The paper is structured as follows: Section 2 of this paper discusses about the standard motion compensation. Section 3 describes least square based prediction algorithm. In this section along with LS, a computationally less complex method (LOPT-3D) is also discussed. In Section 4, the proposed work has been presented. While Section 5 discusses the complexity of the proposed scheme, simulation results are presented in Section. 6. Finally, the conclusions have been drawn in Section 7.

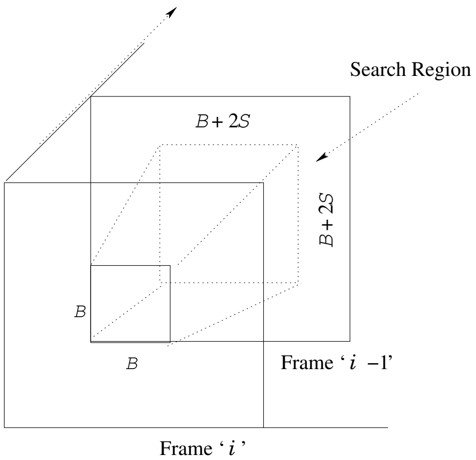

Figure 1: Standard motion compensation

2. MOTION COMPENSATION

Motion compensation is the technique being mostly employed for video compression. In this technique, picture is described in term of transformation of reference picture with reference to the current picture. Compression efficiency improves when images can be accurately synthesized from previously transformed or stored images. There are many motion compensation techniques available in literature like Global motion compensation, Block motion compensation, Variable block size motion compensation, Overlap block motion compensation, Quarter pixel (QPel) and Half pixel motion compensation. This paper uses standard motion compensation for simulation purposes.

Prediction of motion compensated frame in standard motion compensation is done just by using one previous frame. The frame to be predicted is divided in $B \times B$ block (Fig. 1). All of these blocks are non coinciding. For each such block the best match is obtained from the previous frame which serve for the taken block as a motion compensated block. On iterating the above process, the whole motion compensated frames can be constructed. From the methods available in literature for motion compensation, the most common metric used is the absolute difference of the blocks and the minimum of this metric is used to predict the matched block. Its position is called as ‘motion vector’. Like, to find matched block for ‘b_i’ in the i^{th} frame, spatial region of $(B+2S) \times (B+2S)$ in the previous frame is searched. This is known as ‘search region’. Each time, position of the block in the search region is located where the absolute difference if the pixels comes out to be minimum. This is known as motion vector relative to corresponding matched block. The same procedure is followed for each block ‘i’. If $\hat{b_i}$ is the predicted block (i.e. matched block) of b_i then we can write

$$\hat{b_i} = b_{i-1}^{x,y}$$

where x, y is called motion vector belonging to the region $(B+2S) \times (B+2S)$ (search region).

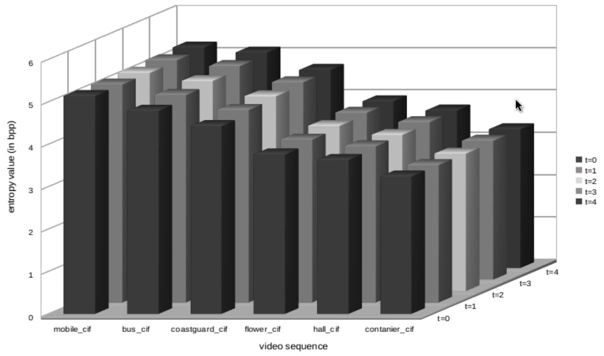

Figure 2: Bit rate(bpp) comparison for QCIF sequences with variation in number of frames skipped (t)

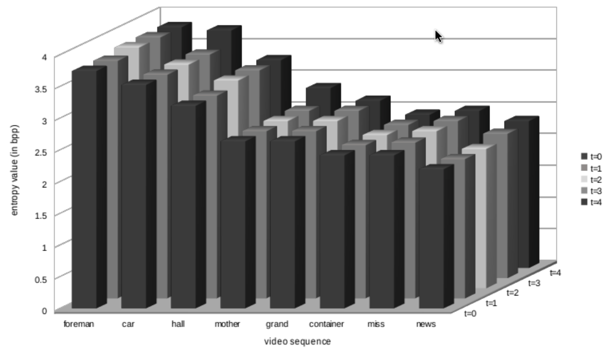

Figure 3: Bit rate(bpp) comparison for CIF sequences with variation in number of frames skipped (t)

.

3. LEAST-SQUARE-BASED PREDICTOR

3.1 Least-Square-Based Predictor for video

This kind of predictor uses training window (W(n) as in Fig. 4(a)) near to the pixel $x(n)$ which is to be predicted, to minimizes the sum of the squared prediction error in that window. The prediction coefficients is then used to predict

Table 1: Variation of percentage of pixels calculating AR parameters with number of frames skipped (t) (QCIF sequences)

Video name	Number of frames skipped (t)				
	0	1	2	3	4
foreman	22.84	13.70	9.27	7.10	5.63
hall	9.20	8.22	5.51	4.15	3.25
car	13.88	8.83	5.95	4.40	3.67
news	5.58	3.77	2.52	1.92	1.77
container	3.31	2.87	1.91	1.46	1.14
grandmother	4.72	3.9	2.56	1.99	1.48
miss	4.61	3.33	2.23	1.72	1.37
mother	6.63	4.80	3.18	2.47	1.89
Average	**8.85**	**6.18**	**4.14**	**3.15**	**2.53**

Table 2: Variation of percentage of pixels calculating AR parameters with number of frames skipped (t) (CIF sequences)

Video name	Number of frames skipped (t)				
	0	1	2	3	4
Container	21.33	10.65	7.14	5.37	4.30
Flower	45.58	22.67	15.28	11.3	9.19
Bus	53.50	26.75	18.10	13.4	13.91
Coastguard	52.23	26.18	17.52	13.1	10.61
Hall	30.81	15.39	10.71	7.75	6.2
Mobile	56.10	28.04	18.84	14.13	11.31

that pixel $x(n)$ giving rise to prediction error $e(n)$ formulated by below.

$$\hat{x(n)} = \sum_{i=1}^{P} a_i x(n-i) \quad (1)$$

where $a(k)$'s are the prediction coefficients and P is the predictor order. In above equation ordering of nearby pixels for video is given in Fig. 4(b), where pixels from previous frame also contribute in prediction of $x(n)$. Prediction coefficients are calculated by minimizing the sum of squared prediction errors which leads to a matrix inversion operation, using training window shown in Fig. 4(a). $x(n)$ is the current pixel and $x(n-i) \quad \forall i = 1, 2, ..., P$ is the pixel in the previous frame and $\hat{X}$ is the predicted value of the pixel $x(n)$.

The procedure to obtain weight of coefficient a_i by using least square based predictor is mentioned below:

$$E = \sum_{k \epsilon S} (x(n) - \hat{x}(n))^2 \quad (2)$$

here, S consists of P neighborhood pixels w.r.t $x(n)$ i.e current pixel for which AR parameters are to be estimated and $\hat{x(n)}$ shows the error estimated by computing the coefficients and then multiplying with the corresponding weights a_i (1). Value of a_i, used to predict the current pixel error are obtained by partially differentiating (2) with respect to each of a_i's. Partial differentiation is carried out to get the minimum value of E (2).

$$\frac{\partial E}{\partial a_i(k)} = 0 \quad \forall i = 1 \textit{ to } 4. \quad (3)$$

The same can be seen in the matrix form as:

$$\sum_{k \epsilon S} D(k) = \sum_{k \epsilon S} X(k) \times A(k)$$

where $D(k)$, $X(k)$, and $A(k)$ are as follows: *

$$D(k) = \begin{bmatrix} x(k-1) \times x(k) \\ x(k-2) \times x(k) \\ x(k-3) \times x(k) \\ x(k-4) \times x(k) \end{bmatrix}$$

$$X(k) =$$

$$\begin{bmatrix} (x(k-1))^2 & x(k-2)x(k-1) & ... & x(k-4)x(k-1) \\ x(k-1)x(k-2) & (x(k-2))^2 & ... & x(k-4)x(k-2) \\ x(k-1)x(k-3) & x(k-2)x(k-3) & ... & x(k-4)x(k-3) \\ x(k-1)x(k-4) & x(k-2)x(k-4) & ... & (x(k-4))^2 \end{bmatrix}$$

$$A(k) = \begin{bmatrix} a_1(k) \\ a_2(k) \\ a_3(k) \\ a_4(k) \end{bmatrix}$$

* Now '$a_i(k)$' can be obtained by matrix inversion of '$A(k)$' and multiplying with '$D(k)$'.

$$A(k) = X(k)^{-1} \times D(k)$$

The same algorithm is used both at encoder and decoder to get the predicted frames. Here rather using the previous frames pixels directly, we use motion compensated frame (as it is more correlated to motion compensated frame) as previous frame for the prediction of current frame pixels. Motion compensated frames are obtained from the method described in Section 2. Thus, we apply LS-based predictor for each pixel to compute to the set of AR parameter for all the predictors. Hence, this method becomes computationally very complex although it gives good compression ratio.

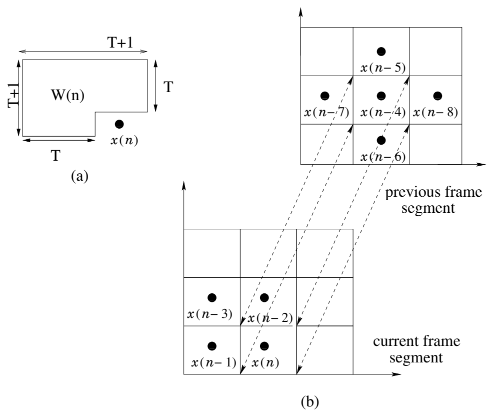

Figure 4: Least-Squares-Based Predictor

3.2 LS-Based Predictor with threshold for video

In a video nearby pixels are mostly correlated. As a consequence they are a probability for them to have same AR parameters. So, to reduce to complexity of LS-based predictor, AR parameters are only calculated for some pixels. The pixels whose AR parameters are being calculated must satisfy the criteria of having previous pixels error $x(n-1)$

Table 3: Comparison of percentage of pixels calculating AR parameters for proposed scheme and LOPT-3D

Video Sequence	Format	LOPT 3D	Proposed Scheme
Container	CIF	21.33	7.14
Flower	CIF	45.58	15.28
Bus	CIF	53.50	18.10
Coastguard	CIF	52.23	17.52
Hall	CIF	30.81	10.71
Mobile	CIF	56.10	18.84
Foreman	QCIF	22.84	9.27
Hall	QCIF	9.20	5.51
Car	QCIF	13.88	5.95
News	QCIF	5.58	2.52
Container	QCIF	3.31	1.91
Grandmother	QCIF	4.72	2.56
Miss	QCIF	4.61	2.23
Mother	QCIF	6.63	3.18

Table 4: Variation of Bit rate(in bpp) with number of frames skipped (t) (QCIF sequences)

Video (frames)	Number of frames skipped (t)				
	0	1	2	3	4
Foreman	3.75	3.75	3.80	3.80	3.80
Hall	3.20	3.20	3.28	3.28	3.28
Car	3.53	3.53	3.53	3.53	3.75
News	2.20	2.20	2.20	2.28	2.28
Container	2.42	2.42	2.42	2.42	2.42
Grandmother	2.64	2.64	2.64	2.64	2.64
Miss	2.42	2.46	2.48	2.48	2.48
Mother	2.64	2.64	2.64	2.64	2.84

greater than certain threshold value (P_{th}).

$$E = x(n) - \sum_{i=1}^{P} a_i x(n-i)$$

$$if E < P_{th}$$

$$\hat{x}(n+1) = \sum_{i=1}^{P} a_i x(n+1-i)$$

$$E = x(n+1) - \sum_{i=1}^{P} a_i x(n+1-i)$$

$$else$$

.

AR parameters are estimated.

.

$$end$$

4. PROPOSED WORK

In this paper, we propose a new LOPT-3D [2] based LS based predictor. This method uses the parameters estimated for a frame, in the prediction of a set of forthcoming frames without any computation. The scheme has been developed

Table 5: Comparison of Bit rate(in bpp) for LOPT-3D and Proposed scheme

Video Sequence	Format	LOPT 3D	Proposed Scheme
Container	CIF	3.25	3.25
Flower	CIF	3.77	3.90
hline Bus	CIF	4.80	4.96
Coastguard	CIF	4.44	4.60
Hall	CIF	3.64	3 69
Mobile	CIF	5.15	5.15
Foreman	QCIF	3.75	3.80
Hall	QCIF	3.20	3.28
Car	QCIF	3.53	3.53
News	QCIF	2.20	2.20
Container	QCIF	2.42	2.42
Grandmother	QCIF	2.64	2.64
Miss	QCIF	2.42	2.48
Mother	QCIF	2.64	2.64

by exploiting the temporal redundancy in consecutive frames as this redundancy has been experimentally observed in the AR parameters. Motion compensation has been applied to use this temporal redundancy to determine the AR parameters for the forthcoming frames without recomputing the LS based parameters. Since, we know the pixel location for which the predictors are estimated, we apply this predictor at the same location in the motion compensated frame. Location wise approximation of predictor is motivated by the fact that the motion compensation has been done with respect to the given frame. The proposed method can be seen as follows.

Let f_k,f_{k+1},.....f_{k+t} be the current frame and the consecutive t frames respectively.

$$A_k = \begin{bmatrix} a_1 \\ a_2 \\ . \\ . \\ . \\ a_P \end{bmatrix}$$

Where A_k be the AR parameters estimated for frame f_k. $\hat{f}_k$,$\hat{f}_{k+1}$,.....$\hat{f}_{k+t}$ are the block based motion compensation frames where the reference frame for motion compensation is taken as f_k ie. the frame for which AR parameters are estimated.

$$\hat{f}_{k+l}(i,j) = f_k^{mc}(i,j)$$

$$\forall i = 1, 2, .., w$$

$$\forall j = 1, 2, ..., h$$

$$\forall l = 1, 2, ..., t$$

Here $f_k(i,j)$ denotes the representation for a particular pixel value at the ($i^{th} and j^{th} location$) represents the frame k, h and w refers to dimension of the frame i.e height and width respectively. f_k^{mc} represents the motion compensated frame for f_{k+l} with reference frame taken as f_k. These $\hat{f}_{k+l}$ frames are more correlated in temporal domain when compared with reference frame f_k. Pixels of the frame $\hat{f}_{k+l}$ are more likely to behave in the same pattern as by frame f_k. So exploiting this stationarity property, we use the same AR parameters on $f_{k+l}(i,j)$ as present at the $f_k^{mc}(i,j)$ position.

Table 6: Variation of storage size (in MB) values with number of frames skipped (t) (CIF formats)

Video (frames)	Number of frames skipped (t)				
	0	1	2	3	4
Container	3.25	3.25	3.25	3.28	3.28
Flower	3.77	3.85	3.90	3.94	3.94
Bus	4.80	4.90	4.96	5.04	5.09
Coastguard	4.44	4.55	4.60	4.66	4.68
Hall	3.64	3.69	3.69	3.72	3.72
Mobile	5.15	5.15	5.15	5.18	5.18

Hence we are not required to estimate the predictors for some consecutive frames and hence saves computational power. However we expect some loss in performance and the same is found to be negligible as compared to the power saving.

5. COMPLEXITY

LS based predictors are too difficult to be computed at the run time in desired time due to limitation of hardware available till date. Least square based predictor are good in performance but lack in implementation due to heavy computational complexity. To overcome this problem a much simpler method was proposed in [2] in which instead all, only for some pixels the parameters are required to be calculated. The pixel for which the AR parameters are estimated are edged ones. A certain threshold value is chosen and if the error comes out to be greater than that threshold P_{th} value then AR parameters are estimated for the next pixels to come, otherwise the same parameters are carried forwarded. This approach tries reducing the computations involved in the process, but even this becomes a challenge for the VLSI engineers to design a very computationally efficient chip. All this reasons together acted as a major backward dragging force for its implementation in the present time.

To overcome this problem, the proposed method presents a low complexity algorithm which holds almost the same performance level. This method estimates the AR parameters for only one frame from a stack of '$t+1$' frames. It clearly shows that the number of computations reduces to $\frac{1}{(t+1)}$ times the originally estimated in the LS predictor. Where as in case of LOPT 3D predictor the same trend is not observed. It can act in the same way with the condition that edged pixels should occur in equal proportion in all the frame of the video sequence, which is very much hypothetical situation. But this is not true for the general case and the AR parameter estimation differs from one frame to the other for a particular video sequence. This trend is shown in Fig. 5 and 6 for QCIF and CIF videos respectively. On an average, fraction of pixels for which AR parameters are needed to be calculated reduces from 43% to 14% in case of CIF video sequences and from 8% to 4% in case of QCIF video sequences. This shows that the proposed method decreases the computations to 3 times for CIF sequences and to 2 times for QCIF video sequences. Table 1 and 2 shows the fraction of pixels for which AR parameter are needed to be estimated for different video sequences. Variable t represents the set of $t+1$ frames, from this set AR parameters are calculated for only one frame and rest of the t frames uses the same for the prediction purpose.

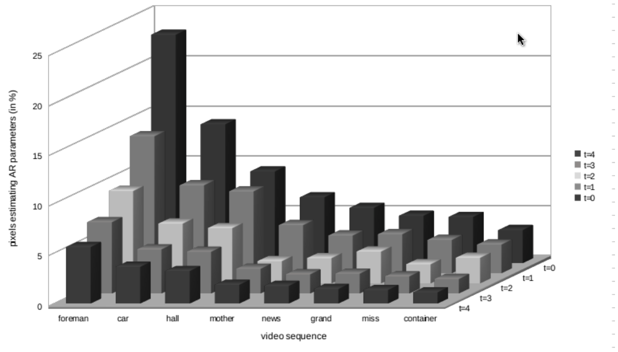

Figure 5: Comparison of average number of pixels for which AR parameters are calculated for different video sequences (QCIF sequences)

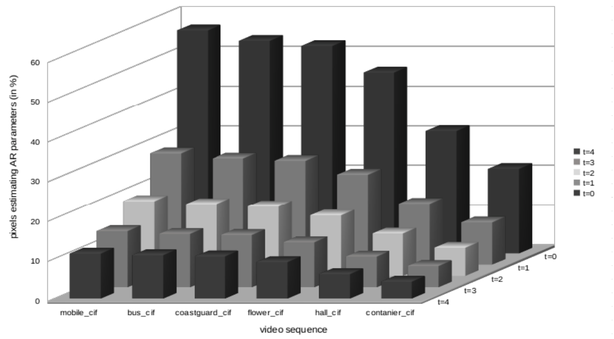

Figure 6: Comparison of average number of pixels for which AR parameters are calculated for different video sequences (CIF sequences)

6. SIMULATION RESULT AND DISCUSSION

Study of our proposed work is done on various standard videos sequences of QCIF and CIF format.

Motivation of the work proposed is to reduce the complexity of LOPT-3D even further without much deterioration in the performance level of the predictor. The LOPT-3D used in this papers takes the threshold value (P_{th}) as 3 as proposed in Dania paper [2]. AR parameters are estimated only if the pixel preceding the current pixel shows the absolute predictor error greater than 3.

Anticipation for doing this was to check the performance of the predictor when the AR parameters are estimated for one frame and the same are used over some of the frames using motion compensation with reference to the frame which estimates the AR parameters. There is a greater chance of finding the edged pixel in the same temporal location as obtained in the reference frame because of block based motion compensation being applied over that reference frame. So using the same AR parameters on the pixels of the current frame to be predicted shall ease the complexity associated with the estimation of the AR coefficients for edged pixels. Even there could have been a chance where the entropy value can decrease on with the variation of t, i.e the set of frames using the same AR coefficients as obtained by the reference frame. There may be some cases for which the predictor assumes the pixel to be an edged ones and estimates the AR coefficients for the non edged pixel. But on using motion compensation the pixel at that temporal location comes out to be an edged pixel and on applying the previously estimated AR parameters the performance may improve. Even if the improvement is not observed but the constant entropy over different values of t is the validation proof for the proposed scheme.

For any low motion video sequences, there is correlation between some successive frames. They show the same behavior as their neighborhood frames. To use this correlation, this method is proposed which estimates the parameters for one frame then use the same for the 't' consecutive frames. Variation in the entropy value with change in 't' is shown in Table. 4 for QCIF sequence, it shows that larger the value of 't' smaller the number of predictors estimated. The same trend is followed in case of CIF video sequence also (Table. 6). With the variation in 't', AR parameters required to be computed also changes. Table. 1 shows this variation in fraction of pixels for which one needs to estimate the AR parameters. This behavior can also be seen graphically from Fig. 6 and 5. By analyzing both the variation i.e. the entropy value and fraction of pixels for which AR parameters are being computed, 't' is chosen. Value of 't' is taken such that neither entropy value becomes too high nor there is negligible difference in the fraction of pixels for which AR parameters are calculated. Keeping in mind this condition, the value of t is chosen as 2.

Now the comparative study of the proposed scheme with the standard LOPT-3D is done. Fraction of pixels for which AR parameters are calculated in proposed scheme and LOPT-3D is shown in Table. 3. It shows that the average number of pixels used in the proposed LOPT-3D is dropped from 8.85% to 4.14% in case of QCIF video sequence. The same can be observed graphically from Fig. 2. Entropy loss is also not much significant, which is 0.07bpp inferior to that of standard which is 2.8064 (Table. 5) for QCIF.

The same procedure is also carried out for the CIF video sequences. Table. 2 and 6 shows the AR parameters estimated and entropy value with the variation in t respectively. Fig. 3 and 6 shows this pattern through graphs. For the chosen value of t i.e $t = 2$, the fraction of pixels reduces from 43% to 14% on an average with the minor loss of .08*bpp* in performance. Arithmetic Coding is used as a entropy coding method in the proposed scheme.

This technique can be used in a much efficient method for video frames which does not change drastically like news reading videos.

7. CONCLUSION

In-spite of having good performance as compared to other predictors, LS based predictors are not used because of its high computational complexity. This paper presents an algorithm which exploits the temporal redundancy property of the video sequence. The proposed scheme estimates the AR coefficients for one frame and apply the same coefficients for the set of t consecutive frames. These t frames are motion compensated with reference to the frame for which we estimates the AR parameters and apply them over the same temporal location. On doing this there is a high probability of getting the edge on the same temporal location or in the neighbourhood of it. This acted as the motivation behind the work proposed and is validated by the entropy values obtained through simulation showing only a minor variation with the change in t. The proposed scheme has been tested over many video sequences for QCIF and CIF format. There has been a significant decrement in the computation cost involved in the LS based predictor with a insignificant loss in the performance level. The computational complexity of estimating AR parameters decreases by a factor of 3 in case of CIF format videos, while in case of QCIF format, decrement by a factor of two has been observed.

8. REFERENCES

[1] R. V. Babu and A. Makur. Object based survelliance video comprssion using foreground motion compensation. In *Proc. ICARCV*, pages 1–6, Singapore, Dec. 2006.

[2] D. Brunello, G. Calvagno, G. A. Mian, and R. Rinaldo. Lossless compression of video using temporal information. *IEEE Trans. on Image Proc.*, 12(2):132–139, Feb. 2003.

[3] E. S. G. Carotti and J. C. D. Martin. Motion compensated lossless video coding in a CALIC framework. In *Proc. ISSPIT*, pages 600–605, December 2005.

[4] D. K. Chobey, M. Vaishnav, and A. K. Tiwari. An optimal switched adaptive prediction method for lossless video coding. In *2013 Data Compression Conference, DCC 2013, Snowbird, UT, USA, March 20-22, 2013*, page 481, 2013.

[5] G. J.Sullivan and R. L. Baker. Motion compensation for video compression using control grid interpolation. In *Proc. ICASSP*, pages 2713–2716, Toronto, Ont. , Canada, May 1991.

[6] D. Lazar and A. Averbuch. Wavelet vidoe compression using region based motion estimation and compensation. In *Proc. ICASSP*, pages 1597–1600, Salt Lake City, UT, May 2001.

[7] N. D. Memon and K. Sayood. Lossless compression of video sequences. *IEEE Trans. Commun.*, 44(10):1340–1345, Octo. 1996.

[8] N.Memon, X.Wo, V.Sippy, and G.Miller. An interband coding extension of the new lossless JPEG standard. In *Proc. SPIE*, pages 1437–1441, Pacific Grove, CA, Jan. 1997.

[9] A. K. Tiwari and R. V. R. Kumar. Least-square based switched adaptive predictors for lossless video coding. In *Proc. ICIP*, volume 6, pages 69–72, San Antonio, TX, Sept. 16-19 2007.

[10] M. Vaishnav, A. Sharma, and A. K. Tiwari. A novel computationally efficient motion compensation method based on pixel by pixel prediction. In *2011 Data Compression Conference (DCC 2011), 29-31 March 2011, Snowbird, UT, USA*, page 480, 2011.

[11] M. Vaishnav and A. K. Tiwari. Bin classification using temporal gradient estimation for lossless video coding. In *Data Compression Conference, DCC 2014, Snowbird, UT, USA, 26-28 March, 2014*, page 430, 2014.