Predictive Block-Matching Motion Estimation for TV Coding -- Part I: Inter-Block Prediction

Sohail Zafar, *Student Member*, IEEE, Ya-Qin Zhang, *Member*, IEEE and John S. Baras, *Fellow*, IEEE

Abstract **- A new motion estimation/compensation scheme for TV coding is proposed in this paper. This scheme is based on prediction of inter-block motion information using minimum absolute difference block matching and hence substantially increases the searching and computation efficiency. A comparison is also made with the "optimal" full motion searching scheme for a standard test sequence.**

I. INTRODUCTION

A variety of data compression schemes have been proposed in the past such as Differential Pulse Code Modulation (DPCM), Discrete Cosine Transform (DCT), Vector Quantization (VQ), Motion Compensation (MC), etc. All compression algorithms capitalize on the redundancies present in natural video signals both in the spatial and the temporal domain. Hybrid schemes using a combination of such techniques have also been applied for intraframe and interframe compression.

Motion compensation or displacement estimation, which intends to obtain the knowledge about the path and speed of the moving objects in a video scene, has been widely applied to various interframe video coding systems. Motion compensation results in a considerable improvement in compression performance as compared to a simple conditional replenishment interframe coding, i.e. coding only the difference of two successive frames. It has been shown that coding of motion compensated frames result in a 25-35 percent reduced data rate as compared to coding of simple frame difference [5] by using a suboptimal approach.

Many motion compensation schemes have been developed recently and can be classified into two categories: block-based pattern matching schemes and pel-based recursive approaches. The block matching approach assumes that all the pels within a block has the same motion activity. This is in contrast to the pel-recursive approach which intends to estimate motion vectors for individual pels. Most of current video communication systems and standards employ a blocking-matching scheme for motion estimation (such as the CCITT proposed px64 video-telephony draft standard and the on-going ISO/CCITT MPEG full-motion video compression draft).

Pel recursive techniques, first introduced by Netravalli and Robins [11], are more representative of the real motion but involve more extensive computations than that for block matching. Block matching techniques were first considered by Rocca, Broofferio et al.[2][3][4] and Limb and Murphy [1]. Although the computational savings associated with block matching techniques as compared to pel recursive method are significant, but still require tremendous computation.

Basically, the factors that determine the performance of a block-based motion estimation scheme are matching criteria, pattern searching schemes and the searching area. Pattern matching criterion defines a measure of pattern distance of motion activity in successive frames. Three types of matching criteria have been defined, namely: maximum cross correlation (MCC), minimum mean square error (MMSE) and minimum absolute difference (MAD). Using these criterion many block-matching searching algorithms have been proposed in the last decade. These include brute-force full search, logarithmic search, 3-step search, conjugate directional search, independent orthogonal search, and more recently the cross-search scheme.

In this paper, we first describe the principles of block-based motion compensation schemes and discuss the merits and drawbacks of different search schemes for motion vector estimation. We will then propose a new searching scheme, namely predictive pattern search (PPS), which utilizes the motion information in the neighboring blocks. Simulation results show that this scheme provides a more efficient search algorithm and potentially provides more realistic motion vectors.

This paper was presented in part at the IEEE Southeastcon'91, Williamsburg, Virginia, April 1991.

Sohail Zafar and John S. Baras are with the department of Electrical Engineering, University of Maryland, College Park MD 20740.

Ya-Qin Zhang is with GTE Laboratories, 40 Sylvan Road, Waltham, MA 02254.

II. BLOCK MATCHING MOTION ESTIMATION

A basic assumption of block-based motion compensation is that the displacement information or motion vectors in the present frame can be efficiently estimated and compensated by searching and matching an optimum matching point in the neighboring frames. In block matching scheme (BMS), a frame of picture is divided into small rectangular blocks at first, assuming that all pels in one block have the same displacement. Then, one of the above mentioned distance function, used as the matching criteria, is calculated around the block in a given search area. The displacement vectors are obtained from the position of the correlation peak or minimum distance.

Figure 1 shows the principle of block-based searching scheme. The picture frame is first divided into many small rectangular blocks each consisting M by N pels. The block at location (m,n) in the ith frame is denoted by $B_i(m,n)$. The motion vectors in horizontal and vertical directions estimated for the $B_i(m,n)$ are denoted by $V_{x,i}(m,n)$ and $V_{y,i}(m,n)$, respectively. The motion compensated residual image, which is called the displaced frame difference (DFD), is denoted by $DFD_i(m,n)$. Clearly, we have:

$$B_{i+1}(m, n) = B_i(m - V_{x,i}(m,n),\ n - V_{y,i}(m,n)) + DFD_i(m,n)$$

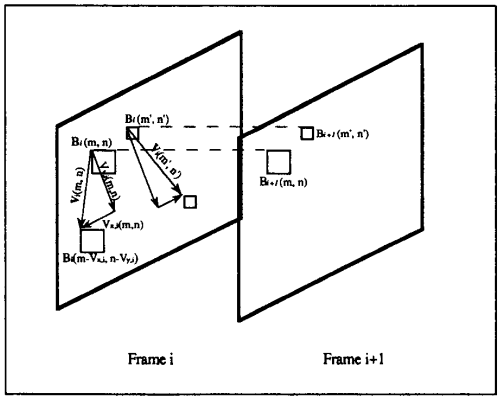

Figure 1: independent block pattern searching scheme

In the independent block pattern searching scheme, the motion vectors for each block are estimated independently, i.e. the $V_{x,i}(m,n)$ and $V_{y,i}(m,n)$ are independent of $V_{x,i}(j,k)$ and $V_{y,i}(m,n)$ for all $m \neq j$ and $n \neq k$. The optimal method is the full search, which as the name suggests, searches each and every location in the specified search area (Ω). But the main disadvantage of this method is that it is

computational very expensive. If the search area Ω is given to be $\pm P$ pels in both the horizontal and the vertical direction, there will be $(2P + 1)^2$ matching calculations for each block in the picture. Furthermore for calculating each match, M by N operations have to be performed. The complexity of these operations depends on the matching criterion, e.g. MCC calculates the cross correlation of the block under consideration with each of the neighboring blocks in the search area.

Thus a block centered at location (m, n) will have the motion vectors $V_{x,i}(m,n)$ and $V_{y,i}(m,n)$, if the block (m,n) has the lowest distance with respect to the matching criterion. Note that $(m,n) \in \Omega$ is the area scanned by locations $[k\pm P, l\pm P]$ area around pel (k,l) with intensity I(k,l). MCC is calculated for each block located at (m,n) as below.

$$\mathbf{V}_i(m,n) = \arg \max_{x,y \in \Omega} \left\{ \frac{\sum_{p=-M/2}^{M/2} \sum_{q=-N/2}^{N/2} I_i(k+p, l+q)\, I_{i+1}(x+p, y+q)}{\sqrt{\left(\sum_{p=-M/2}^{M/2} \sum_{q=-N/2}^{N/2} I_i^2(k+p, l+q)\right) \left(\sum_{p=-M/2}^{M/2} \sum_{q=-N/2}^{N/2} I_{i+1}^2(x+p, y+q)\right)}} \right\}$$

The term in the numerator is the cross correlation and the two terms in the denominator are the autocorrelation of the two blocks. The next less computational expensive matching criterion is the Minimum Mean Squares Error (MMSE) which calculates the mean squared error for each location in the search area and then chooses the position of the block which is the closest in the mean squared sense

$$\mathbf{V}_i(m,n) = \arg \min_{x,y \in \Omega} \left\{ \frac{1}{MN} \sum_{p=-M/2}^{M/2} \sum_{q=-N/2}^{N/2} \left(I_i(k+p, l+q) - I_{i+1}(x+p, y+q) \right)^2 \right\}$$

The simplest of all the matching criterion is the Minimum Absolute Difference criteria, which calculates the absolute difference for every block in the neighborhood and then chooses the one with the minimum value as the match.

$$\mathbf{V}_i(m,n) = \arg \min_{x,y \in \Omega} \left\{ \frac{1}{MN} \sum_{p=-M/2}^{M/2} \sum_{q=-N/2}^{N/2} \left| I_i(k+p, l+q) - I_{i+1}(x+p, y+q) \right| \right\}$$

MAD is the most widely used matching criterion of all due to its lower complexity and good performance. It has been shown that MCC does not necessarily give the best results in terms of data rate. In fact it is very hard to model the motion vector prediction errors analytically. The characteristics depend very much on the scene itself i.e., the contents of the scene, motion of the objects in the scene, etc.

Some sub-optimal searching schemes do not scan the whole search area but still produce a bit rate reasonably close to that from the full search. Many such schemes have been studied in the past. A brief description of these is presented here for comparison. For the details the reader is encouraged to lookup in the referenced articles [1] through [10].

The 2-D directed search method introduced by Jain and Jain [5] is an extension of the binary or logarithmic search in one dimension (binary sort). The search area is successively reduced at each step. At every step the five locations which contain the center of the area, the mid-points between the center and the four boundaries of the area along the axis passing through the center are searched. This procedure continues until the plane of search reduces to only 3x3. In the final step all the nine locations are searched and the location of the minimum distortion is selected as the motion vectors. The distortion measure is the mean squared error. This algorithm reduces the number of calculations from $(2P + 1)^2$ required by the full-search to only $(2 + 7\log P)$.

The three-step motion vector direction search algorithm presented by Koga et al. [6] first coarsely searches the specified area and then reduces the granularity of search at each step. The distortion measure is a non-linear function. This scheme results in a dramatic reduction of operations to just $(1 + 8\log P)$ for a search size of P in both horizontal and vertical directions. The results show that the entropy difference between the results using this scheme and the full search was small as compared to the prediction error entropy of the interframe prediction.

Another search algorithm for the direction of the motion vectors is based on conjugate direction presented by Srinivasan and Rao [7]. Starting at the center of the block the vertical direction is kept fixed while the horizontal direction is varied to find the minimum of the distortion. From this minimum location the horizontal is kept constant while the vertical is varied to find the minimum in the vertical direction. This method is also called the one-time-search (OTS) algorithm. The maximum number of searches is thus (2P + 3) and results in only an insignificant degradation in peak-to-peak signal-to-noise ratio.

The Orthogonal Search algorithm (OSA) was introduced by Puri et al. [8] in which, with a logarithmic step size, at each iteration four new locations are searched. In this method, at every step, two new positions are searched alternately in the vertical and horizontal direction. Thus the total number of searches is reduced to $(1+4\log_2 P)$.

Recently, the Cross-Search algorithm for motion estimation was introduced by Ghanbari [9]. In this algorithm, the basic idea is still the logarithmic step search as described by Jain [5] and Koga et al.[6], with the main difference that, at each step there are four search locations which are the end points of the diagonals rather than the centers of the vertical and horizontal. Thus the total number of positions searched are $(5+4\log_2 P)$. A comparison of computational complexities of the existing algorithms is also given by Ghanbari [9] which indicates that the orthogonal search is the fastest followed by cross-search algorithm.

III. PREDICTIVE PATTERN SEARCH

There are many types of motion activities in natural video signals. One of the objectives of motion-compensation is to compensate for the translation caused by object moving or camera panning the scene. In scenes with this type of motion, a large homogeneous area of the picture frame, which may consists of many blocks (we call it a superblock), is very likely to move in the same direction with similar velocities. Therefore, the motion vector information within a superblock (e.g. a cluster of 4 by 4 blocks) is highly correlated or dependent. The main point behind the proposed Predictive Pattern Search (PPS) scheme is to take advantage of this correlation and hereby reduce the searching area. It is pointed out here that freshly uncovered areas due to motion of an object have very little correlation to its neighbors in terms of motion vectors, and these areas should not be coded by inter-frame schemes.

In the proposed inter-block PPS scheme shown in Fig. 2, the motion vector $\mathbf{V}_{i,1}(m,n) = (V_{x,i,1}(m,n), V_{y,i,1}(m,n))$ of the first block in a superblock is calculated by the full-search scheme. Then the motion vectors of the adjacent blocks within the superblock are estimated by,

$$\mathbf{V}_i(m',n')\} = E[\mathbf{V}_i(m,n)] + \mathbf{d}_i(m,n) \quad \text{for } |m' - m| \le P \text{ and } |n' - n| \le P$$

where $|\mathbf{d}_i| \le D$ are the motion vectors calculated with a reduced search area D which is much smaller than that for the first block (e.g. one quarter). The initial estimate $E\{\mathbf{V}_i(m, n\}$ can be found using a two-dimentional causal autoregressive (AR) model, represented as:

$$E\{\mathbf{V}_i(m,n)\} = \sum_{p,q \in \Theta} \alpha_{p,q}\, \mathbf{V}_i(m - p, n - q)$$

Where $\{a_{p,q}\}$ is a set of prediction coefficients for each block within a superblock and Q is a causal set defined by:

$$\Theta = \{ p, q: q > 0 \; \forall \; p \} \cup \{ p, q: q=0, p > 0 \}$$

The motion vectors are refreshed from superblock to superblock. Therefore, estimation error is limited and doesn't propagate to another superblock. It should be pointed out that reduced motion

searching area also provides an additional compression since the correlation of motion vectors is reduced (which means the overhead information of motion vectors is less).

In our simulations, we have used a very simple model for estimation, given by:

$$E\{\mathbf{V}_{i,s}(m,n)\} = (V_{x,i,1}(m,n)+d_{x,i,1}(m,n),\ V_{y,i,1}(m,n)+d_{y,i,1}(m,n))$$

where $|d_{x,i,1}| \le D_x$ and $|d_{y,i,1}| \le D_y$ are the motion vectors calculated with a reduced search area and the superblock is a 2 by 2 cluster.

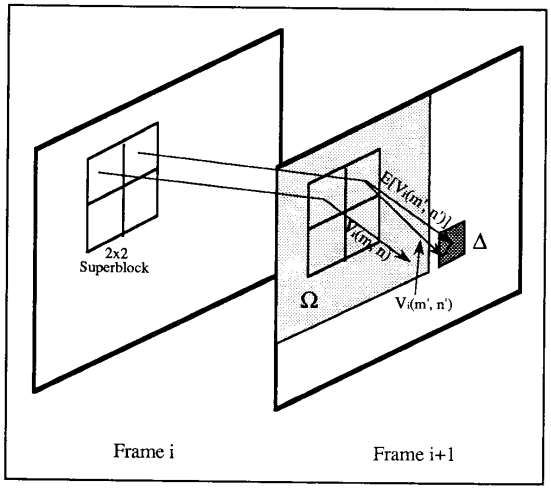

Figure 2: Inter-block Predictive pattern searching scheme

IV. Inter-Block Predictive Motion-Compensated Video Compression System

The PPS is implemented on an Abekas video compression testbed set up in GTE Laborotories, Waltham, MA as shown in Figure 3.

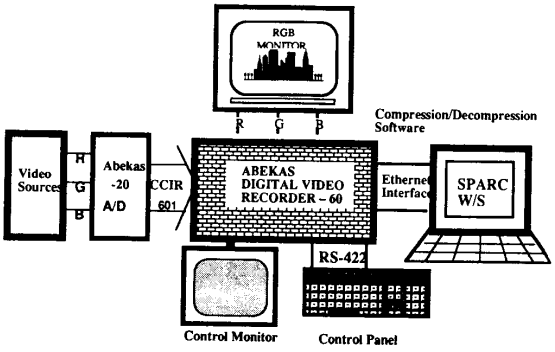

Figure 3. Configuration of the TV Coding Testbed in GTE Lab., MA

An interframe hybrid DCT-based video compression scheme was implemented for a full-motion test sequence "CAR" using different motion compensation schemes.

The Abekas A-60 is basically a digital video recorder which allows a real-time playback of 25 second recorded CCIR 601 digital video. The Abekas system is interfaced with a Sparc workstation via Ethernet. The compression software resides on the Sparc. By software simulation of the compression/decompression techniques in the Spark, we can reconstruct video segments to compare with the original signal via a real-time play back. Therefore, we can evaluate compression performance, quality degradation, computational efficiency of different coding algorithms associated with various motion compensation techniques.

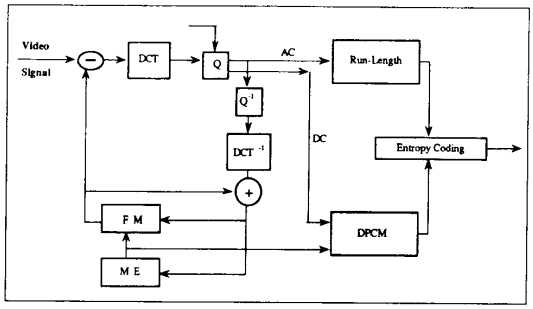

Figure 4. Block Diagram of Inter-frame motion-compensated hybrid DCT/DPCM video compression scheme

As shown in Figure 4, the compression scheme we implemented is an interframe hybrid DCT/DPCM scheme motion compensated by inter-block predictive motion compensation. After motion compensation, the displaced difference frame is block DCT transformed and uniformly quantized. The quantization process takes advantage of human visual characteristics and coefficients are weighted prior to the uniform quantization. Then, the AC coefficients are zig-zag scanned and run-length coded. The DC coefficients of adjacent blocks are further DPCM coded to reduce the inter-block redundancy. Motion vectors are noiselessly coded and transmitted. All quantities are entropy-coded prior to transmission.

The "CAR" is a full-motion color video sequence in CCIR 601 format with 720 by 480 per frame, 16 bits per pixel in YUV format (or 24 bits in RGB format) and 30 frames/60 fields per second. It is basically a fast camera panning sequence which is ideal for testing different motion compensation algorithms.

V. Experimental Results

Figure 5 (a) and (b) show the probability distribution of motion vectors in vertical and horizontal directions for the second frame. FMS denotes the motion vectors calculated by full search and the numerical value immediately following signifies the search area. For predictive search PMS16 motion vectors are estimated from the first block of the 2 by 2 superblock. The searching area for full

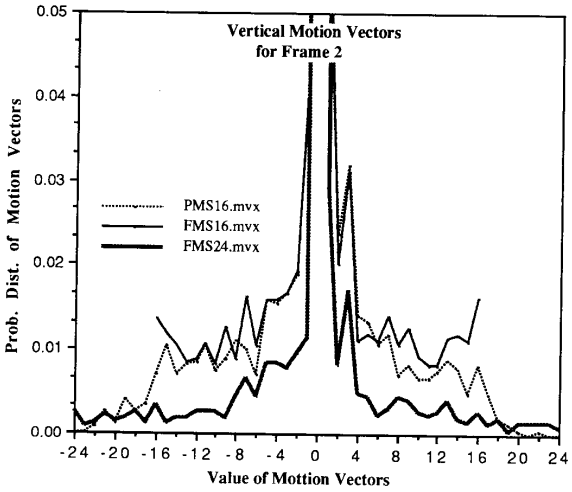

Figure 5 (a). Probability Distribution of motion vectors in the vertical direction for "CAR" sequence. Searching area is 16x16 for FMS16 and PMS16, 24x24 for FMS24 and reduced D for predictive search is 4x4.

search is 16 by 16 for FMS16 and 24 by 24 for FMS24. The search area for the first block within a superblock for predictive search is 16 by 16 while the reduced search area D for the rest of the blocks is 4 by 4. The extreme values generated at a value of zero have not been shown to emphasize the lower probability values. It is noted that the motion vector distribution in vertical direction is very close for the two schemes. However, the estimated motion vectors in horizontal direction differ a lot. This is due to the nature of heavy motion in the horizontal direction in the "CAR" sequence. In this case, the full search scheme FMS16 is limited by its maximum searching size (16 by 16), where predictive search tends to be more flexible and can provide a more realistic measure of motion activities. This can also be observed from figure 5(b) that the horizontal motion vectors for FMS16.mvy are bounded at the value 16. For predictive search PMS16.mvy, there is also a false peak at the value 16, which is caused by the implementation of refreshing within a superblock rather than the predictive search algorithm. The real motion peak is around 20, which is reflected by the full search scheme FMS24.mvy where the search area is 24 by 24. Certainly, false peak can be avoided by increasing the searching area at the expense of heavy computation.

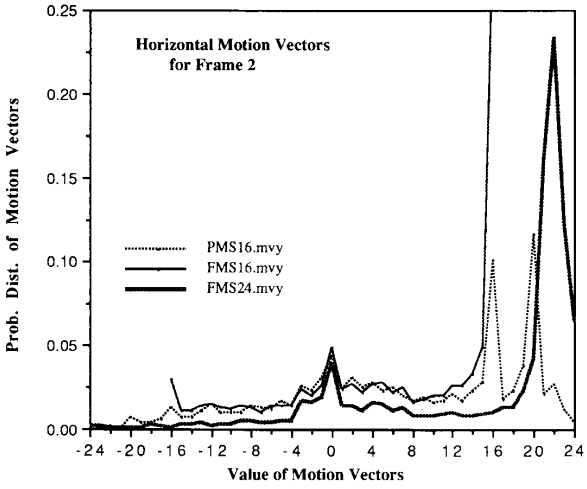

Figure 5 (b). Probability Distribution of motion vectors in the horizontal direction for "CAR" sequence. Searching area for FMS16 and PMS16 is 16x16, 24x24 for FMS24 and reduced D for predictive search is 4x4.

Motion compensation greatly improves the coding efficiency of a video compression system. One parameter to measure the improvement is the entropy, which defines the coding limit for a noiseless compression scheme. Although the compression scheme used in this work is irreversible, the entropy still provides some insights into the coding limit and criterion to compare different motion compensation schemes. The first-order entropies of displaced frame difference (DFD) are illustrated in Figure 6 for coding with no motion compensation, full-search motion compensation and predictive search motion compensation, respectively. Clearly, motion compensation decreases the sample-measured entropy by a factor of 20 to 30 percent, and entropy result of predictive search is a little bit higher than that of the full search compensation.

In an actual motion-compensated video compression system, not only the coded DFD information is transmitted/stored, the displaced information, i.e. the motion vectors, have also to be transmitted/stored in order to correctly retrieve the motion displacement factor at the decoding end. Therefore, the entropy of motion vectors is also a critical measure of the compression efficiency. The first-order sample-measured entropies are depicted in Figure 7 for the motion vectors in both vertical and horizontal directions for full-search and inter-block predictive-search motion compensations, respectively. Clearly, the predictive-search significantly reduces the entropy of the motion vectors. As a matter

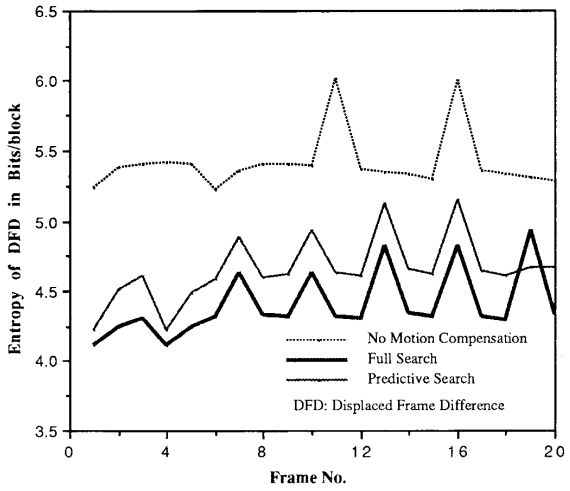

Figure 6. Entropy of Displaced Frame Difference in Bits/Pixel for No Motion Compensation (zero displacement), Full-search Motion Compensation and Inter-Block Predictive Motion Compensation, respectively.

of fact, only half of the overhead needs to be transmitted in the predictive scheme. This is an obvious advantage of the proposed inter-block predcitive-search scheme.

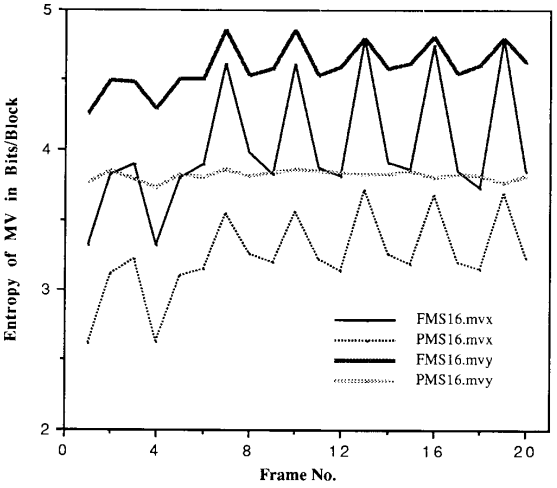

Figure 7. Entropy of Motion Vectors in Vertical & Horizontal Direction in Bits Per Block (8 by 8) for Full Search and Inter-block Predictive Search Motion Compensation.

The effect of motion compensation can also be measured by the signal-to-noise ration, which is defined as:

$$\frac{S}{N}\,(\mathrm{dB}) = 10 \log_{10} \frac{(\text{Peak-to-peak Value of the Signal})^2}{\text{Variance of Reconstruction Error}}\ (\mathrm{dB})$$

Figure 8 shows the signal-to-noise ratios for 20 reconstructed "CAR" frames which are hybrid coded by DCT/DPCM and motion-compensated by full search and predictive search schemes, respectively. It can be easily concluded from Figure 8 that the overall signal-to-noise ratio of the predictive searching scheme is comparable to that of the full search schemes. Actual coding results from 800 kbps to 3 Mbps are displayed in the Abekas video system in GTE Video Lab. However, the computational burden for

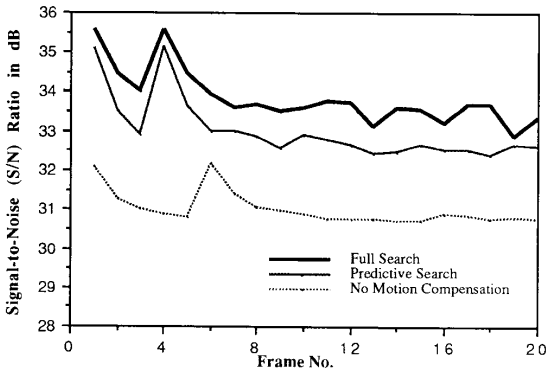

Figure 8. The reconstructed Signal-to-Noise Ratio in dB for Full Search and Inter-Block Predictive Search Schemes, respectively.

predictive searching is reduced as much as four times since the searching area is reduced.

VI. CONCLUSIONS

A new motion estimation/compensation scheme for full-motion interframe video compression was proposed in this paper. This scheme is based on prediction of inter-block motion information using minimum absolute difference block matching and hence substantially increases the searching and computation efficiency. A comparison has also been made with the "optimal" full motion searching scheme for a standard test sequence. The experimental results show that this scheme provides a computational efficient algorithm in a real-time video communication environment. It should be pointed out that these results are obtained for inter-block predictive search motion compensation, studies in inter-frame predictive search and hybrid inter-block/inter-frame motion compensation will be reported in the accompanying paper [12].

REFERENCES

[1] J.Limb and J.Murphy, "Measuring the speed of moving objects from TV Signals," *IEEE Trans.Commun.*, Vol.COM-23, pp.474-478, April 1975
[2] F.Rocca and S.Zanoletti, "Bandwidth reduction via movement compensation on a model of the random video process," *IEEE Trans. Commun.*, Vol.Com-20, pp.960-965, Oct. 1972
[3] C.Cafforio and F.Rocca, "Method for measuring small displacements of TV images," *IEEE Trans. Inform. Theory*, Vol.IT-22, pp. 573-579, Sept. 1976
[4] S.Brofferio and F.Rocca, "Interframe redundancy reduction of video signals generated by translating objects," *IEEE Trans. Commun.*, Vol.Com-25, pp.448-455, April 1977
[5] J.Jain and A.Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commu.*, Vol.Com-29, No.12, Dec. 1981
[6] T.Koga, K.Iinuma, A.Hirano, Y.Iijima and T.Ishiguro, "Motion-compensated interframe coding for video conferencing," *Proc. NTC'81*, pp.G5.3.1 -G5.3.5
[7] R.Srinivasan and K.Rao,"Predictive Coding based on efficient motion estimation," *IEEE ICC'84*, pp.521-526, 1984
[8] N.Ohta, M.Nomura and T.Fujii, "Variable Rate Coding Using Motion-Compensated DCT for Asynchronous Transfer Mode Network," *IEEE ICC'88*, pp.1257-1261, 1988
[9] M.Ghanbari, "The cross-search algorithm for motion-estimation," *IEEE Trans. Commun.*, Vol.38, No.7, pp.950-953, July, 1990
[10] H.Musmann, P.Pirsch and H.Grallert, "Advances in picture coding," *Proc. of IEEE*, Vol.73, No.4, pp.523-548, April 1985
[11] A.Netravili and J.Robbins, "Motion compensated television coding - part I," *Bell Syst. Tech. J.*, Vol.58, pp.631-670, 1979
[12] Y.Zhang and S.Zafar, " Predictive Block-Matching Motion Estimation for TV Coding--Part II: Inter-Frame Prediction, " *IEEE Trans. Broadcasting, this issue*

Sohail Zafar was born in Lahore, Pakistan, on November 3, 1960. He received his B.Sc. Degree in Electrical Engineering from University of Engineering and Technology, Lahore, Pakistan, in 1981, and his M.S. from Columbia University, New York, NY in 1988. Since 1989, he has been working as a Graduate Research Assistant at the University of Maryland, College Park, Maryland, where he is pursuing his Ph.D. He has worked as Member of Technical Staff at Contel Technology Center, Chantilly, VA during the summer of 1989 and 1990. He is working as a summer Member of Technical Staff at GTE Laboratories, Waltham, MA. His research interests include Neural Networks, Parallel Processing and Video Coding and Transmission.

Ya-Qin Zhang received his B.S. and M.S in Electrical Engineering from China University of Science and Technology (USTC), Hefei, China, in 1983 and 1985, respectively. He received his Doctor of Science (Sc.D) in Electrical Engineering from George Washington University, Washington, D.C in 1989. He is a Senior Member of Technical Staff at GTE Laboratories, Waltham, MA. He was a Member of Technical Staff and later as a Senior Member of Technical Staff at Contel Technology Center, Chantilly, VA. He was on the faculty of Taiyuan University of Technology in Shanxi, China in 1986 and became a part-time faculty at George Washington University in 1990. He has published more than 30 papers in image/video coding and transmission, and medical imaging. He received the Merwin PhD award co-sponsored by ten major communications companies for his academic achievements in 1989. He is a member of IEEE and Eta Kappa Nu.

John S. Baras is a professor at the Deparmrnt of Electrical Engineering at the University of Maryland, College Park, MD.