VIDEO CODING BASED ON ITERATED FUNCTION SYSTEMS

Baldine-Brunel Paul and Monson H. Hayes III

Georgia Institute of Technology
School of Electrical and Computer Engineering
Atlanta, Georgia 30332-0250

ABSTRACT

An approach is presented for the compression of motion video sequences using Iterated Function Systems. In the proposed approach, the video stream is partitioned into three-dimensional range regions. Each range region consists of a variable number of rectangular blocks that belong to consecutive frames along motion trajectory. Our approach exploits correlation between consecutive blocks in the direction of motion by predicting the IFS map of a given range block with that of a parent range block along the trajectory of motion. The proposed approach shows good promise for efficient modeling and compression of motion video sequences at an affordable computational cost.

1. INTRODUCTION

This paper addresses the problem of modeling motion video sequences using Iterated Function Systems for the purpose of compression. Iterated Function Systems or IFS have been shown to efficiently model real world images in [1], [2], [3], [4]. Early results in fractal based video compression were reported in [5] and [6]. The IFS coding paradigm consists of representing the signal to be modeled as an approximation to the fixed point of a set of contractive mappings forming an iterated function system. The typical approach is to partition the original signal, image or video, into a set of non-overlapping target blocks or *range* blocks of pixel values and to associate a contractive transformation with each range block.

In the approach we propose, the video stream is partitioned into three-dimensional range regions. Each range region consists of a variable number of rectangular blocks that belong to consecutive frames along the same motion trajectory. Range regions may vary in depth due to occlusion and uncovered background near the boundaries of moving objects. The IFS map for the first block in any range region is found by performing a full search for the best domain block in the corresponding frame. For each of the remaining blocks in the range region, the IFS map is found by performing a limited search around the projection of the domain block associated with the previous range block along the motion trajectory. In this case, some of the parameters that describe the IFS map for the first block can be propagated to the following blocks in the range region.

Our approach exploits correlation between consecutive blocks in the direction of motion by predicting the IFS map of a given range block with that of a parent range block along the trajectory of motion. We are in effect extending the assumption of near intensity constancy along motion trajectory to IFS maps. Since frame to frame motion is relatively limited in scope, it is computationally more efficient to perform motion estimation for a matching range block in a previous frame, followed by a refinement of the predicted map in the current frame.

This work was supported in whole or in part by the Joint Services Electronics Progrm, Grant No. DAAH-04-93-G-0027

2. BACKGROUND

For a given range block $\boldsymbol{r}_i$, the affine transformation on the pixel values has the general form:

$$\boldsymbol{r}_i = \sum_{j=1}^{N} \alpha_j \boldsymbol{f}_j + \sum_{k=1}^{P} \beta_k \boldsymbol{d}_{ik} \tag{1}$$

where $\{\boldsymbol{f}_j\}_{j=1}^{N}$ represents a set of N fixed vectors of pixel values in the sense that the elements of these vectors are image independent. The vectors $\{\boldsymbol{d}_{ik}\}_{k=1}^{P}$, referred to as *domain* blocks, are image dependent blocks associated with the range block $\boldsymbol{r}_i$. These domain blocks are chosen from a pool generated in the following manner. The original image is subsampled in the horizontal and vertical directions by some factors σ_H and σ_V respectively. Blocks of the same size as the range blocks are extracted from this subsampled image to generate an intermediate pool of domain blocks Ω_{int}. The final

0-7803-2431-5/95 $4.00

pool of domain blocks Ω_D is generated from that intermediate pool by shuffling the pixel elements within each intermediate domain block according to some predetermined pattern. This shuffling referred to as *isometry* by Jacquin [1],[7] is a form of reindexing of the pixels within the block as investigated by Vines [8]. Each resulting domain block in the pool Ω_D is characterized by its index or coordinates in the subsampled image and the specific isometry that was used to generate that domain block.

It is customary to refer to the mapping specified by Equation 1 as a *massic* or grey-scale transformation, since it involves operations on the pixel values. In contrast, the operations involved in generating the pool of domain blocks constitute an underlying contractive spatial transformation due to the reduction in the original image size. This insures that the resulting IFS will be contractive at least in the spatial domain. In effect, the affine transformation described by Equation 1 maps a larger block from the original image onto a range block within the same image, hence the concept of ***self-transformability*** or ***self-similarity*** by affine transformation or ***self-affinity***.

The objective of an IFS image modeling algorithm is to find, for each range block $\boldsymbol{r}_i$ in the original image, a corresponding set of domain blocks $\{\boldsymbol{d}_{ik}\}_{k=1}^{P}$ such that the approximation

$$\hat{\boldsymbol{r}}_i = \sum_{j=1}^{N} \alpha_j \boldsymbol{f}_j + \sum_{k=1}^{P} \beta_k \boldsymbol{d}_{ik} \qquad (2)$$

is as close as possible to the given range block $\boldsymbol{r}_i$. Let $D_i \in \Omega_D^P$ be the set of optimal domain blocks associated with the range block $\boldsymbol{r}_i$; that is, let $D_i = \{\boldsymbol{d}_{ik}\}_{k=1}^{P}$; then D_i is the solution to the following minimization problem:

$$D_i = \arg \min_{D \in \Omega_D^P} ||\boldsymbol{r} - \hat{\boldsymbol{r}}||^2 \qquad (3)$$

where the Euclidian norm is used for convenience. Then the parameters that specifies the IFS map are appropriately quantized and encoded for transmission ar storage.

On the decoder side, the input consists of a set of affine maps specified by the parameters: $\{\alpha_i\}_{i=1}^{N}$, $\{\beta_k\}_{k=1}^{P}$, $\{\boldsymbol{x}_k\}_{k=1}^{P}$, $\{h_k\}_{k=1}^{P}$ where $\boldsymbol{x}_k$ represents the coordinates of the domain block $\boldsymbol{d}_{ik}$ in the reduced image and h_k specifies the corresponding isometry. Starting with an arbitrary image, the attractor of the iterated function system is sought by using the following iteration, for each range block $\boldsymbol{r}_i$ in the image:

$$\boldsymbol{r}_i^{(s)} = \sum_{j=1}^{N} \beta_j \boldsymbol{v}_j + \sum_{k=1}^{P} \alpha_k \boldsymbol{d}_{ik}^{s-1} \qquad (4)$$

In this expression, $\boldsymbol{r}_i^{(s)}$ represents the i^{th} range block at iteration s in the reconstruction process.

3. FIXED SUBSPACE PROJECTION METHOD

The IFS maps used in this work are in terms of a set of N fixed basis vectors $\{\boldsymbol{f}_n\}_{n=1}^{N}$ and one domain vector in the following form:

$$\boldsymbol{r} = \sum_{j=1}^{N} \alpha_j \boldsymbol{f}_j + \beta \boldsymbol{d}_o \qquad (5)$$

In this expression, the vector $\boldsymbol{d}_0$ is the projection of the domain vector $\boldsymbol{d}$ on the subspace orthogonal to the fixed basis vectors and is given by:

$$\boldsymbol{d}_o = (\boldsymbol{I} - \boldsymbol{P_F})\boldsymbol{d} \qquad (6)$$

The projection matrix $\boldsymbol{P_F}$ onto the subspace $\mathcal{F}$ is given by :

$$\boldsymbol{P_F} = \boldsymbol{F}\left(\boldsymbol{F}^T\boldsymbol{F}\right)^{-1}\boldsymbol{F}^T \qquad (7)$$

where:

$$\boldsymbol{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_N] \qquad (8)$$

This approach is an extension of the inner product space method proposed in [9] where a single fixed vector was used for the IFS map.

Equation 5 can be rewritten as:

$$\boldsymbol{r} = \boldsymbol{F\alpha} + \beta \boldsymbol{d}_o \qquad (9)$$

where $\boldsymbol{\alpha}$ is the N-dimensional vector consisting of the coefficients of the fixed vectors. For a given domain vector $\boldsymbol{d}$ and a range vector $\boldsymbol{r}$, we can generally find least-squares solutions for $\boldsymbol{\alpha}$ and β in Equation 9. Specifically, $\boldsymbol{\alpha}$ is the least-square solution to the following equation:

$$\boldsymbol{F\alpha} = \boldsymbol{r} \qquad (10)$$

that is:

$$\boldsymbol{\alpha} = \left(\boldsymbol{F}^T\boldsymbol{F}\right)^{-1}\boldsymbol{F}^T\boldsymbol{r} \qquad (11)$$

under the condition that the matrix $\boldsymbol{F}^T\boldsymbol{F}$ is invertible. The resulting expression for β is given by:

$$\beta = \frac{<\boldsymbol{d}, \boldsymbol{r} - \boldsymbol{F\alpha}>}{<\boldsymbol{d}, \boldsymbol{d} - \boldsymbol{F\gamma}>} \qquad (12)$$

where the N-dimensional vector $\boldsymbol{\gamma}$ has the form

$$\boldsymbol{\gamma} = \left(\boldsymbol{F}^T\boldsymbol{F}\right)^{-1}\boldsymbol{F}^T\,\boldsymbol{d} \qquad (13)$$

and represents, in general, the least square solution of the following equation:

$$\boldsymbol{F\gamma} = \boldsymbol{d} \qquad (14)$$

Based on this formulation, the collage approximation is given by:

$$\hat{\boldsymbol{r}} = \boldsymbol{F\alpha} + \beta[\boldsymbol{d} - \boldsymbol{F\gamma}] \qquad (15)$$

The corresponding collage error is given by:

$$\boldsymbol{e} = [\boldsymbol{r} - \boldsymbol{F\alpha}] - \beta[\boldsymbol{d} - \boldsymbol{F\gamma}] \qquad (16)$$

4. IFS MODELING OF VIDEO

4.1. Domain Vector Prediction

Temporal redundancy can be exploited in various ways for IFS-based video compression. In one approach, the concept of *near-constancy* of pixel intensity in the direction of motion may be translated into near-constancy of the map parameters in the direction of motion. Let $\boldsymbol{r}_{i,k}$ and $\boldsymbol{r}_{i+\Delta i,k+1}$ be two range blocks taken from consecutive frames k and $k+1$; the second range block results from a Δi increment of the index due to motion. It is expected that the IFS map that describes the range vector $\boldsymbol{r}_{i,k}$ will be *close*, in the mean-squared sense, to the IFS map that describes the translated range block. In particular let $\boldsymbol{d}_{i,k}$ be the domain vector associated with range block $\boldsymbol{r}_{i,k}$. The domain vector associated with range block $\boldsymbol{r}_{i+\Delta i,k+1}$ is then predicted using the displaced domain block $\boldsymbol{d}_{i+\Delta i,k+1}$ in the image frame $k+1$ as illustrated in Figure 1. By using this notion of near-constancy of the map parameters as a constraint in the optimization process, the search for the best domain block is confined to a limited neighborhood around the predicted domain block in a predicted frame, assuming that motion estimation was successful for the range block under consideration. Significant reduction in the search complexity has been achieved as reported in [6]. This approach is certainly more suitable for intermediate bit-rate coding.

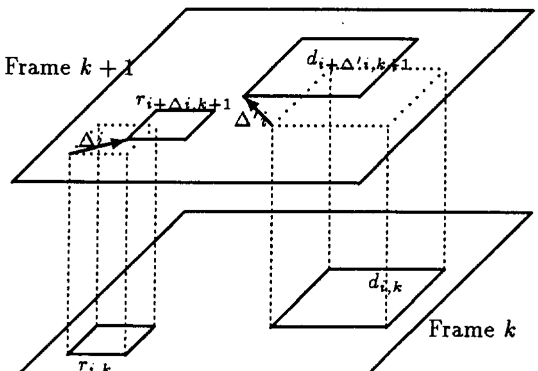

Figure 1: Domain vector prediction for IFS video coding

4.2. Modified Motion Compensated Prediction of Range Blocks

For applications requiring very low bit rates, we investigated the following approach. The video sequence is partitioned into predicted frames and non-predicted frames. In a predicted frame, a range block may be predicted or coded independently. The collage approximation of a motion-compensated range block in a predicted frame is given by:

$$\hat{\boldsymbol{r}}_{i,k} = \boldsymbol{r}_{i-\Delta i,k-1} + \beta\boldsymbol{d} \qquad (17)$$

That is, the fixed component of the IFS map is taken to be the motion-compensated prediction $\boldsymbol{r}_{i-\Delta i,k-1}$ of the current range block $\boldsymbol{r}_{i,k}$. For a non-predicted block, the IFS map is formulated using the fixed subspace projection method; that is:

$$\hat{\boldsymbol{r}}_{i,k} = \boldsymbol{F\alpha} + \beta\boldsymbol{d}_o \qquad (18)$$

where $\boldsymbol{\alpha}$ is given by Equation 11. A decision criterion can be formulated to select between the two modes of operation. Furthermore, in the case of a motion-compensated block, another decision criterion may be formulated as whether to set $\beta = 0$ in Equation 17. This decision must be done adaptively by weighing the decrease in distortion that is contributed by the domain vector term versus the increase in bit allocation for that particular range block.

5. SIMULATION RESULTS

Our preliminary results are illustrated in the following figures and tables. Simulations were performed using of the test sequence *Miss America*. The frame size for this sequence is 176×144. The motion estimation range was set to ± 7. For the spatial transformation, the image is subsampled by a factor of 2 in both directions to generate the pool of domain blocks. We also use a total of 8 isometries on the decimated domain blocks. In this particular simulation, a new intraframe is coded at the beginning of each second of video. This corresponds to an intraframe period of 30 frames. The remaining frames are coded as predicted frames with an average of 96% of the range blocks coded with motion compensation. The maximum value for β was 0.95 in the simulations. With a target bit-rate of 128 Kbits/sec, the frame rate of the input sequence is 30Hz. The resulting actual bit-rate is about 112 Kb/s. The corresponding average signal-to-noise ratio is near 35dB.

These results indicate that Iterated Function Systems can be used to efficiently encode motion video sequences at very low bit-rates. Additional work is currently done to improve the quality of the decoded images and to achieve compression at lower bit rate.

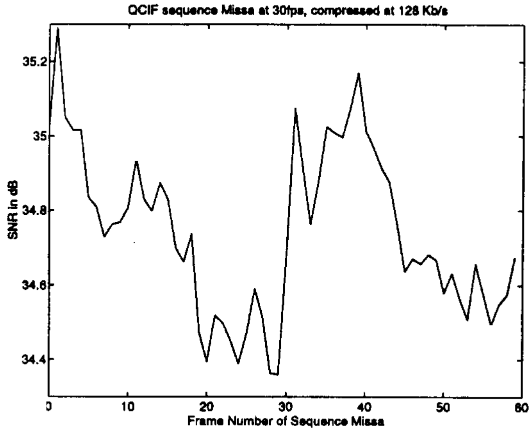

Figure 2: SNR versus Frame Number of QCIF sequence Missa: actual bit rate is 112 Kb/s, frame rate is 30Hz, and frame size is 176 x 144

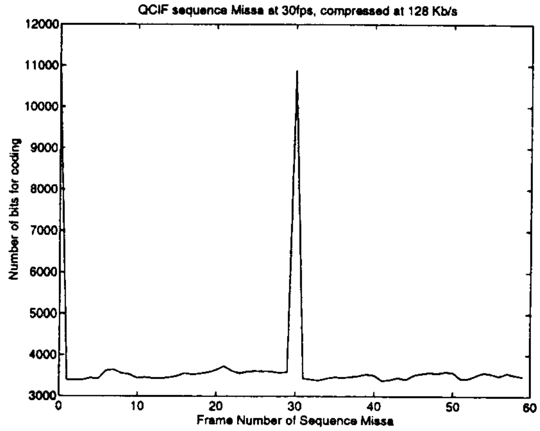

Figure 3: Coded Bits versus Frame Number of QCIF sequence Missa: actual bit rate is 112 Kb/s, frame rate is 30Hz, and frame size is 176 x 144

6. REFERENCES

[1] A. E. Jacquin, "A novel fractal block-coding technique fo digital images," in *ICASSP*, pp. 2225–2228, 1990.

[2] G. Vines and M. H. Hayes, "Orthonormal basis approach to ifs image coding," in *Proceedings of IEEE Multidimensional Signal Processing Workshop*, September 1993.

[3] D. M. Monro, "A hybrid fractal trasform," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP93*, vol. 5, pp. 169–172, April 1993.

[4] M. Gharavi-Alkhansari and T. Huang, "A fractal-based image block coding algorithm," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP93*, vol. 5, pp. 345–348, 1993.

[5] D. L. Wilson, J. A. Nicholls, and D. M. Monro, "Rate buffered fractal video," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP94*, vol. 5, pp. 505–508, April 1994.

[6] B.-B. Paul and M. H. Hayes, "Fractal-based compression of motion video sequences," in *Proceedings of International Conference on Image Processing, ICIP94*, pp. 755–759, November 1994.

[7] A. Jacquin, *A Fractal Theory of Iterated Markov Operators with Applications to Digital Image Coding*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 1989.

[8] G. Vines, *Signal Modeling with Iterated Function Systems*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, 1993.

[9] G. E. Oien, S. Lepsoy, and T. A. Ramstad, "An inner product space approach to image coding by contractive transformations," in *Proceedings of International Conference on Acoustics, Speech and Signal Processing, ICASSP91*, vol. 4, pp. 2773–2776, 1991.