MOTION TRANSFORMS FOR VIDEO CODING

Dinei A. F. Florêncio and Robert M. Armitano and Ronald W. Schafer

Digital Signal Processing Laboratory
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, GA 30332
floren@eedsp.gatech.edu armi@eedsp.gatech.edu rws@eedsp.gatech.edu

ABSTRACT

Video coding standards use motion compensated prediction to reduce temporal redundancies. In general this is performed by computing motion vectors for predefined regions in the image (rectangular blocks), and transmitting these vectors as side-information. The Motion Transform (MT) [1] provides a new motion compensation technique that does not require the transmission of motion vectors and yet, performs as well as forward motion estimation for many image sequences. In the MT, motion information is obtained in a bottom-up hierarchical fashion using a two-dimensional non-linear filter bank. In this paper we elaborate on MT implementation details and discuss a new variation of the MT. Results obtained using MTs on typical image sequences are compared to results for the full search block matching algorithm (BMA).

1. INTRODUCTION

Motion estimation/compensation is an important component in the video coding paradigm, since motion compensated prediction efficiently reduces temporal redundancies that exist in typical image sequences. In video coding standards (e.g. MPEGI, MPEGII and H.263 [2, 3]), motion compensation is used in a predictive coding scheme to compress the input image sequence. In low bit-rate applications a significant portion of the available channel bandwidth is occupied by motion side-information. The Motion Transform (MT) was recently proposed [1] to circumvent the need for transmitting motion vector side-information and, in doing so, lowering the bit-rate. Motion information is obtained in a coarse-to-fine hierarchical decomposition using non-linear filter banks [4]. In this paper, we extend the work presented in [1]. Further results are given that illustrate the effectiveness of the MT in video coding schemes.

The MT uses a pyramidal decomposition with a non-linear filter bank [4]. Expansiveness is eliminated by directly applying an association of the filter bank and perfect reconstruction is guaranteed by imposing conditions on the filter bank structure [4, 5]. The video coder uses this hierarchical decomposition to remove several levels of spatial redundancy. Non-linear decomposition allows motion matching between different resolution images. Initially the lowest resolution representation of the frame is coded and transmitted to the decoder. At the encoder and decoder the reconstructed lowest resolution frame is used as the "current frame" to compute displacement vectors for the next higher level in the image hierarchy. Any previously transmitted frame can be used as a reference. Using motion compensated prediction, the residual signal, at the current resolution level, is computed. The residual is then intra-frame coded and transmitted to the decoder. By adding the predicted frame to the residual, the representation of the frame at the given resolution level is reconstructed. Refinement of increasingly higher resolution motion vectors is possible, as the decoder and encoder work up through the image hierarchy. Motion estimation is performed at each resolution level to provide the most accurate predicted frame, until the full resolution image is recovered.

The MT can technically be classified as a backward motion estimation technique, but it performs as well as the BMA without motion vector transmission. In addition by reconstructing the image in an hierarchical fashion, progressive transmission is possible. In [1] block based MT was implemented. In this paper an overlapped pixel based motion estimation routine is explored.

In Section 2 we describe the MT. Simulation results are given in Section 3 for a number of different MPEG input sequences. In Section 4 the conclusions are presented.

2. THE MOTION TRANSFORM

The MT is a hierarchical motion-based decomposition of a frame, based on one or more reference frames. Taking the place of traditional motion compensation techniques in a video coder, it reduces temporal redundancy in image sequences. Different from most motion compensation techniques, MTs do not make use of side-information to transmit motion information, relying instead on information that is available at both the encoder and decoder.

MTs have many characteristics that are common to traditional spatial transforms and filter banks. First, the decomposition is non-expansive (e.g. the total number of samples in the transformed domain is equal to the number of samples in the original frame). Second, it relies on common previous knowledge (the basis functions in traditional

This work supported in in part under the Joint Services Electronics Program, Contract DAAH-04-93-G-0027, by CNPq (Brazil) under contract 200.246-90/9 and by grants from the Hewlett-Packard Company.

0-7803-3258-X/96/$5.00 © 1996 IEEE

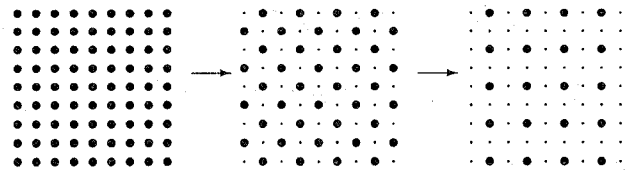


Figure 1: **Converting between rectangular and quincunx grids by subsampling.**

transforms, and the reference frames in the MTs). The image is decoded (i.e., inverse transformed) from the transform samples using this common knowledge base. On the other hand, the MTs are non-linear transforms; the inverse transform is not obtained by a linear combination of basis functions. It is obtained via a complex non-linear relation with the reference frames, that attempts to recover the relative displacement between the pixels in the current frame and the pixels in the reference frame.

Figures 2 through 4 illustrate the basic idea of a motion transform. First, versions of the current frame at several resolutions are obtained. A simple and effective way of obtaining these reduced resolution version is by direct subsampling of the original signal. The lowest resolution image is transmitted to the decoder, using a simplified video compression scheme. The lowest resolution representation of the frame needs to be finely quantized since every resolution level is dependent on the lower level. On the other hand, the lowest level representations require fewer bits to code due to their limited resolution. Starting from the lowest resolution representation, the higher resolution images are successively interpolated by considering the motion in relation to the reference frame. Interpolation error is transmitted to the decoder (this error is similar to the motion compensated residual in the BMA and is shown in Figure 5). The next resolution level estimates motion based on the transmitted residual and the next higher resolution image from the reference frame.

The process is iterated until the final, full resolution, image is obtained. The whole process can be analyzed as a critically decimated nonlinear filter bank decomposition, using the framework proposed in [4].

2.1. A Specific MT

Specific implementations of the MT will have unique forms of producing the lower resolution images and the motion compensated residual. In the MT example implemented in this paper, the reduced resolution versions of the image are obtained by simple subsampling, and the motion compensation is an overlapping BMA scheme with integer pixel accuracy and illumination compensation.

The subsampling scheme follows a 2:1 ratio at each stage. For best results, this 2:1 subsampling is done using a quincunx subsampling grid, instead of using a row/column elimination strategy. A quincunx sampling grid is equivalent to a rectangular grid tilted by 45°, and therefore after a second subsampling we obtain a rectangular grid again. Figure 1 illustrates this process.

To exploit the fact that no side-information is used

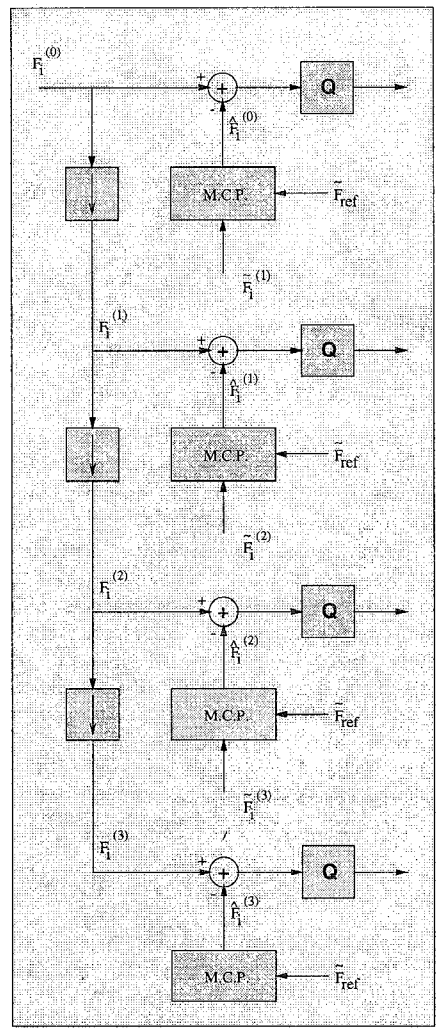


Figure 2: **Motion Transform Encoder.**

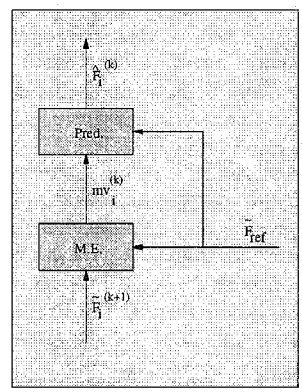


Figure 3: **Motion Compensation Block.**

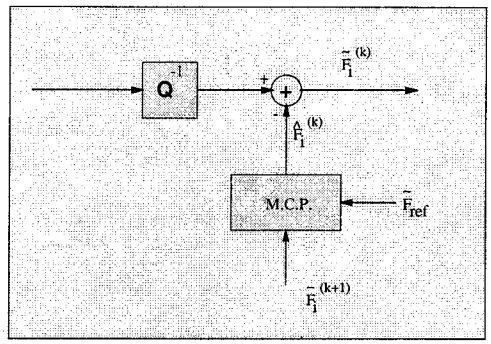


Figure 4: **Motion Transform Decoder.**

by the MTs, we use small, highly overlapping blocks to compute pixel-by-pixel motion displacements in the current frame. Motion vector side-information is not transmitted, removing constraints on the number (or type) of motion vector used during motion compensation. Each pixel displacement is computed using a (4×4) block that is centered on the pixel of interest. Motion displacement is then computed for that block. The displacement vector becomes the motion vector for the pixel an the center of the current block. In this manner, we compute one motion vector and one illumination compensation coefficient for each pixel in the image. Computing motion vectors on a pixel-by pixel basis is computationally expensive. Similarly, traditional block matching can be used to reduce the computational load as proposed in [1].

To simplify notation, let us ignore the quincunx sampling grid, and use a 4:1 subsampling scheme. Denote by F_i^0 the full resolution image at time i. At each stage of the decomposition a lower resolution image is obtained by subsampling the current image by a total factor of four; i.e.,

$$F^{(k+1)}[n,m] = F^k[2n,2m]. \quad (1)$$

This spatial decomposition is performed repeatedly until the lowest resolution image is obtained.

To code the lowest resolution image F_i^K an estimate, $\hat{F}_i^K$, is obtained using motion compensated prediction. Motion information can be obtained using linear predictive coding or in the example in this paper, the lowest resolution motion vector is set to zero, and the predicted frame $\hat{F}_i^K = \tilde{F}_{ref}^K$. In other words, the frame difference is coded and sent to the receiver.

For subsequent levels in the hierarchy a lower resolution image is always available for motion estimation. At level k, decoded image $\tilde{F}_i^{k+1}$, is used as the current frame for motion estimation, as shown in Fig. 3. Any frame that was previously transmitted can be used as the reference frame, $\hat{F}_{ref}^0$. Since $F^{(k+1)}$ is a subsampled version of $F^{(k)}$ some samples are common for both. An estimate for each non-common sample $F^{(k)}[n,m]$ is obtained by selecting the samples of $F^{(k+1)}$ contained in a small window around around $F^{(k)}[n,m]$ and searching for the best match in samples contained in the search region in the reference frame $\hat{F}_{ref}^0$. Note that the samples in the reference frame should be taken at the same sampling distance as those in $F^{(k)}$, but the accuracy of the motion vector can be made equal to one pixel in the full resolution image without any need for interpolation. If the relative displacement between the two blocks that showed the best match is $[h,v]$, and if $[nn,mm]$ is the position of $F^{(k)}[n,m]$ in the full resolution image, then $\hat{F}_{ref}^0[nn+h,mm+v]$ is used as an initial estimate for $F^{(k)}[n,m]$.

This predicted block is further enhanced by illumination compensation. This is done by adding to $F^{(k)}[n,m]$ the difference between the average of the four pixels surrounding $F^{(k)}[n,m]$ and the average of the corresponding pixels on the "best match". The predicted signal, $\hat{F}_i^k$, is then used to form the motion compensated residual signal that is subsequently transmitted to the receiver.

At the receiver the process is reversed as shown in Fig. 4.

3. RESULTS

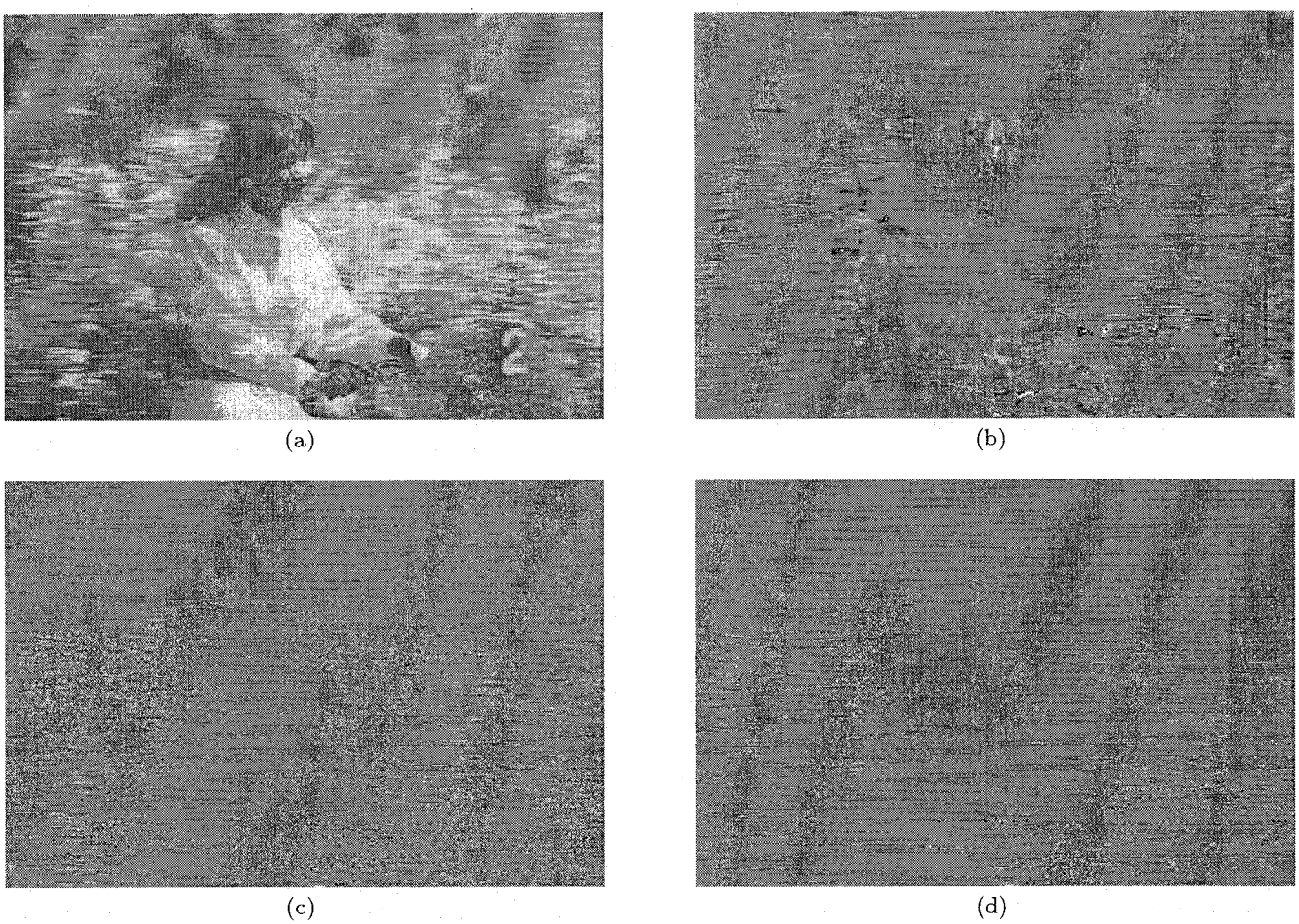
Results for the MT are shown with (8×8) and (16×16) search regions. The MT implementation depicted uses highly overlapped (4×4) blocks and a three level hierarchical decomposition. The performance of the MT is illustrated in Figure 5 where the residual signal is plotted for the MPEG test sequence, `CycleGirl`. Quantitatively the MT is compared to the BMA in Table 1 for a number of MPEG test sequences (SIF-NTSC) with a resolution of 352×240 pixels.

The interpolation error of the MT is compared to the motion compensated residual for Frame 52 of the `CycleGirl` sequence in Figure 5. The original frame is shown in Figure 5.a. The motion compensated residual for the block matching algorithm using a (16×16) search region is shown in Figure 5.b. The MT is shown in Figure 5.c with a (8×8) search region with overlapping blocks. The MT is shown in Figure 5.d using a (16×16) and overlapping blocks. The MT residual signal is similar in nature to the motion compensated residual, but is not blocky. This may be unfavorable when block-based intra-frame coding (e.g., DCT based) is used to encode the residual signal.

Sequence	BMA (16×16) (dB)	MT (8×8) (dB)	MT (16×16) (dB)
`CycleGirl`	23.15	22.85	24.93
`Flamingo`	24.24	25.91	26.54
`FlowerGarden`	21.11	22.05	24.59
`Football`	27.80	31.91	33.65
`TableTennis`	26.80	28.05	29.12

Table 1: **Performance comparison between the BMA and the MT.**

In Table 1 the results for five video test sequences are shown for the BMA and the MT. Only one frame per sequence is depicted in the table, however, in all cases the performance is measured at frame number 52. The performance measure used is the PSNR of the predicted frame (when compared to the original frame) and is shown in dB. In all five test sequences, the MT with a (16×16) search



(a) (b) (c) (d)

Figure 5: **Motion Transform Example**

region outperforms the full search BMA with equivalent search sizes. In all but one case (the `CycleGirl` sequence) the MT with a (8×8) search region will perform better than the BMA with a search region that is four times as large.

4. CONCLUSIONS

The MT is a motion compensation scheme that does not require motion vector side-information and is suited for progressive signal transmission. The MT uses a pyramidal decomposition with a non-linear filter bank. Expansiveness is eliminated by directly applying an association of the filter bank and perfect reconstruction is guaranteed by imposing conditions on the filter bank structure. The video coder uses this hierarchical decomposition to remove several levels of spatial redundancy. Non-linear decomposition allows motion matching between different resolution images. The MT performs well when compared to the BMA and does not require motion vector overhead. On the other hand, computation of pixel displacement must be performed at both the encoder and decoder, requiring symmetric computational engines.

In this paper a MT with highly overlapped blocks was compared to the full search BMA. The MT performed better than the BMA without transmitting motion vectors.

5. REFERENCES

[1] Robert M. Armitano, Dinei A. F. Florêncio, and Ronald W. Schafer. The motion transform: a new motion compensation technique. 1996.

[2] Vasudev Bhaskaran and Konstantinos Konstantinides, editors. *Image and Video Compression Standards.* Kluwer Academic Publishers, 1995.

[3] Hans Georg Musmann, Peter Pirsch, and Hans-Joachim Grallert. Advances in picture coding. *Proceedings of the IEEE*, 73(4):523–548, April 1985.

[4] Dinei A. F. Florêncio and Ronald W. Schafer. Perfect reconstructing non-linear filter banks. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 1996.

[5] P. P. Vaidyanathan. *Multirate Systems and Filter Banks.* Prentice-Hall, 1993.